

# Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory

Reza Olfati-Saber  
Control and Dynamical Systems  
California Institute of Technology  
e-mail: olfati@cds.caltech.edu

June 22, 2004

*Submitted to the IEEE Transactions on Automatic Control*

## Abstract

In this paper, we present a theoretical framework for design and analysis of distributed flocking algorithms. Two cases of flocking in free-space and presence of multiple obstacles are considered. We present three flocking algorithms: two for free-flocking and one for constrained flocking. A comprehensive analysis of the first two algorithms is provided. We demonstrate the first algorithm embodies all three rules of Reynolds. This is a formal approach to extraction of interaction rules that lead to the emergence of collective behavior. We show that the first algorithm generically leads to regular fragmentation, whereas the second and third algorithms both lead to flocking. A systematic method is provided for construction of cost functions (or collective potentials) for flocking. These collective potentials penalize deviation from a class of lattice-shape objects called  $\alpha$ -lattices. We use a multi-species framework for construction of collective potentials that consist of flock-members, or  $\alpha$ -agents, and virtual agents associated with  $\alpha$ -agents called  $\beta$ - and  $\gamma$ -agents. We show that the tracking/migration problem for flocks can be solved using an algorithm with a peer-to-peer architecture. Each node (or macro-agent) of this peer-to-peer network is the aggregation of all three species of agents. The implication of this fact is that “flocks need no leaders”. We discuss what constitutes flocking and provide a universal definition of “flocking” for particle systems that has the same role as “Lyapunov stability” for nonlinear dynamical systems. By “universal”, we mean independent of the method of trajectory generation for particles. Various simulation results are provided that demonstrate the effectiveness of our novel algorithms and analytical tools. This includes performing 2-D and 3-D flocking, split/rejoin maneuver, and squeezing maneuver for 40 to 150 agents (e.g. particles and UAVs).

**Keywords:** flocking, swarming, distributed algorithms, self-organizing networks, collective behavior, cooperative control, multi-vehicle systems, graph Laplacians, dynamic graphs, mathematical biology

## 1 Introduction

Flocking is a form of collective behavior of large number of interacting agents with a common group objective. For many decades, scientists from rather diverse disciplines including animal behavior, physics & biophysics, social sciences, and computer science have been fascinated by the emergence of flocking/swarming/schooling in groups of *agents* via local interactions [47, 38, 37, 36, 29, 40, 53, 51, 48, 26, 14, 52, 34]. Examples of these agents include birds, fish, penguins, ants, bees, and crowds.

The engineering applications of flocking include massive mobile sensing in an environment; parallel and simultaneous transportation of vehicles or delivery of payloads; and performing military missions such as reconnaissance, surveillance, and combat using a cooperative group of Unmanned Aerial Vehicles (UAVs). Flocking is an example of a coordinated task performed by dynamic agents over (visually invisible) self-organized networks in nature. The self-organizing feature of flocks/schools [34] can provide a deeper insight in design of sensor networks [5, 28, 13, 8, 1].

Particle-based flocking is also one of the elements of 3-D animation technology that has revolutionized the entertainment industry. This success is partly owe to the pioneering work of Reynolds [40].

# Report Documentation Page

Form Approved  
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>22 JUN 2004</b>		2. REPORT TYPE		3. DATES COVERED <b>00-00-2004 to 00-00-2004</b>	
4. TITLE AND SUBTITLE <b>Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>California Institute of Technology, Control and Dynamical Systems, Pasadena, CA, 91125</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited</b>					
13. SUPPLEMENTARY NOTES <b>The original document contains color images.</b>					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES <b>37</b>	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

In 1986, Reynolds introduced three heuristic rules that led to creation of the first computer animation of flocking [40]. Here are three quotes from [40] that describe these rules:

- 1) “Flock Centering: attempt to stay close to nearby flockmates,”
- 2) “Obstacle Avoidance: avoid collisions with nearby flockmates,”
- 3) “Velocity Matching: attempt to match velocity with nearby flockmates.”

Let us mention that these rules are also known as *cohesion*, *separation*, and *alignment* rules in the literature. The main problem with implementation or analysis of the above rules is that they have broad interpretations. The issue of how to interpret Reynolds rules was resolved after publication of more recent papers by Reynolds [41], [42].

Among the first groups of physicists who studied flocking from a theoretical perspective were Vicsek *et al.* [53], Toner & Tu [51], and Shimoyama *et al.* [48]. The work of Vicsek *et al.* was mainly focused on emergence of alignment (not flocking) in a particle system, whereas Toner & Tu used a continuum mechanics approach. Shimoyama *et al.* proposed a particle-based model that is capable of describing various forms of swarming behaviors using a network with all-to-all interconnections (i.e. a centralized algorithm). Other continuum models of alignment/swarming were proposed by Mogilner and Eldstein-Keshet [25, 26] with consideration of the sensing issues.

Recently, there has been a great interest among control scientists in analysis of alignment phenomenon and consensus problems. This includes the alignment results in networks with switching topologies by Jadbabaie *et al.* [16] and Olfati-Saber & Murray [33]. The results in [16] were further extended to networks with directed topologies by Moreau [27] and Ren & Beard [39]. The convergence analysis of a nonlinear consensus/alignment protocol was first presented in [45]. Helbing *et al.* [14] studied the *escape panic phenomenon* in which a crowd that is trapped in a large building (or arena) attempts to evacuate that location in case of an emergency. In Liu *et al.* [22], stability analysis of one-dimensional swarms with a fixed topology is studied. In [17], Justh & Krishnaprasad proposed an algorithm for swarming that requires all-to-all interconnections and is only analyzed for the case of two agents. Gazi & Passino [11] employed social potentials to study cohesion in swarms using a centralized mechanism. Tanner *et al.* in [50] proposed a centralized algorithm for a particle system that leads to *irregular collapse* for generic initial states. They also proposed a distributed algorithm that leads to *irregular fragmentation*<sup>1</sup>. Fragmentation and collapse are two well-known pitfalls of flocking. These phenomena do not constitute flocking in the sense described in Section 9.

Some past research with strong connections to this paper include the work by the following researchers: Fax & Murray [9] used graph Laplacians for multi-vehicle formation stabilization; Mesbahi [23, 24] emphasized the role of state-dependent graphs in information networks (interestingly, “dynamic flocks” turn out to be examples of state-dependent graphs); Cortes & Bullo [5] and Cortes *et al.* [6] introduced distributed algorithms for placement of mobile sensors that appears to be closely related to flocking in a constrained space; Rabichini & Frazzoli [43] proposed an energy-efficient method for determining the split time in a multi-vehicle rejoin/split maneuver; Leonard & Fiorelli [21] used artificial potentials for formation keeping; Ogren *et al.* [28] use potential functions for mission planning and coordination of mobile sensor networks; Strogatz [49] elucidates the features of complex networks in physical, biological, and social networks and suggests analytical exploration of complex networks with dynamic topologies; Olfati-Saber & Murray [32] used potential functions induced by rigid graphs for formation control; and Khatib [18] and Rimon & Koditschek [44] used artificial potentials for obstacle avoidance.

A number of recent papers on flocking/swarming, [17, 50, 11], make a set of assumptions that lead to algorithms with some (or all) of the following properties:

- a) The algorithms are centralized and require that each agent interacts with every other agent.
- b) The algorithms do not possess (environmental) obstacle avoidance capabilities.
- c) The algorithms lead to irregular fragmentation and/or collapse.
- d) Unbounded forces are used for collision avoidance.
- e) The algorithms do not possess distributed tracking (or migration) capabilities for groups.

---

<sup>1</sup>By “irregular fragmentation”, we mean disintegration of a flock into smaller groups combined with severe violation of inter-agent constraints. Later, we elaborate on features of fragmentation and collapse and their differences with flocking.

Centralized algorithms for flocking/swarming are neither scalable in terms of computational and communication costs (i.e. the total number of links), nor feasible from a biological perspective because the sensing range of an individual agent is limited.

To the best of our knowledge, there are only two classes of scalable flocking algorithms that do not possess properties a) through e) and successfully lead to creation of flocking in presence of obstacles. Namely, the work by Chang *et al.* [4] that uses gyroscopic forces for collision avoidance and the work by Olfati-Saber [31] (summarized in [46]) which uses multi-species collective potential functions for both imposing inter-agent constraints and obstacle avoidance. So far, there are no analytical results that relate the algorithm in [4] to flocking. In [31] some analytical results such as asymptotic alignment and energy dissipation are established. The analysis presented in [31] and some other existing works on flocking are far from complete. In particular, the existence of a spatial order in flocks/schools has never been established. In this paper, we attempt to bridge some of these theoretical gaps.

From the perspective of control theory, complex systems, and animal behavior, there are a number of fundamental questions that have inspired this work:

1. How do we design scalable flocking algorithms and guarantee their convergence?
2. What does cohesion mean for groups and how is it achieved in a distributed way?
3. What are the stability analysis problems related to flocking?
4. What types of order exist in flocks?
5. How do agents in flocks perform obstacle avoidance?
6. How do flocks perform split/rejoin maneuvers or pass through narrow spaces?
7. How do flocks migrate from point A to B? do they need any leaders?
8. What is a flock? and what constitutes flocking?

We address questions 1–5 in the context of flocking in free-space and questions 6 and 7 as we discuss flocking in presence of environmental obstacles. Finally, at the end of the paper, we discuss what constitutes flocking. We hope that our analysis sheds light on mechanisms of cooperation and emergence of collective behavior in groups.

Let us refer to a particle/agent in a group with the objective of performing flocking as an  $\alpha$ -agent. We introduce three scalable flocking algorithms for  $\alpha$ -agents. Our first algorithm is a gradient-based algorithm equipped with a velocity consensus protocol. We demonstrate that the first algorithm embodies all three rules of Reynolds. It is also demonstrated that this algorithm leads to regular fragmentation rather than flocking for generic initial states. The analysis of the first algorithm is very useful for clarifying the features of the regular fragmentation phenomenon. The second algorithm (or Algorithm 2) is the main flocking algorithm for moving in a free  $m$ -space. This algorithm has an additional term represented by a  $\gamma$ -agent that takes the group objective into account. In the process of analyzing Algorithm 2, we pose two conjectures that are crucial in explaining the spatial order of flocks and self-assembly of a connected network of mobile agents.

The third algorithm (or Algorithm 3) has obstacle avoidance capabilities. We represent the effects of obstacles via virtual agents called  $\beta$ -agents. These agents are kinematic and move on the boundary of the obstacles. A multi-species collective potential is then formed that is used for both design and analysis of the third flocking algorithm. We demonstrate that the tracking problem for flocks can be solved using a distributed algorithm with a peer-to-peer architecture. This analytically confirms a widely accepted opinion by animal behavior scientists that “schools need no leaders” [47].

The main contribution of this paper is to provide a theoretical and computational framework for creation and analysis of scalable flocking algorithms in  $\mathbb{R}^m$  in presence (or lack) of environmental obstacles. Two important pitfalls of flocking are identified and avoided. The tracking problem for groups is addressed. A universal definition of flocking is provided for particle systems that is similar to the notion of “Lyapunov stability” for nonlinear systems. The definition is “universal” in the sense that it does not depend on the method of trajectory generation for particles. A verification test for flocking is proposed based on plotting four curves obtained from the trajectory of particles. We provide several simulation results for 40 to 150 agents that successfully perform 2-D flocking, 3-D flocking, 2-D regular fragmentation, split/rejoin maneuver, and squeezing maneuver in a distributed manner.

An outline of the paper is as follows: Some background on graphs, nets,  $\alpha$ -lattices, algebraic graph theory, consensus problems, collective potentials are presented in Section 2. Two distributed flocking algorithms for moving

in free-space are provided in Section 3. Collective dynamics of flocks and a decomposition lemma are stated in Section 4. Our main results on analysis of the first two algorithms are presented in Section 5. The relation between Algorithm 1 and Reynolds rules are established in Section 6. The third algorithm with obstacle avoidance capabilities is presented in Section 7. Extensive simulation results are provided in Section 8. In Section 9, we elaborate on what constitutes flocking and give a universal definition of flocking. Finally, concluding remarks are made in Section 10.

## 2 Preliminaries

The theoretical framework presented in this paper for both creation and stability analysis of flocking behavior relies on some basic concepts in graph theory [2, 7, 15], algebraic graph theory [12], spatially induced graphs [31], and consensus problems [45, 33] that are discussed in the following.

### 2.1 Graphs and Nets

A *graph*  $G$  is a pair  $(\mathcal{V}, \mathcal{E})$  that consists of a set of vertices  $\mathcal{V} = \{1, 2, \dots, n\}$  and a set of edges  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ . The quantities  $|\mathcal{V}|$  and  $|\mathcal{E}|$  are, respectively, called *order* and *size* of the graph. In the context of networked dynamic systems,  $|\mathcal{E}|$  is referred to as the *communication complexity* of a network with information flow  $G$  [19].

The *adjacency matrix*  $A = [a_{ij}]$  of a graph is a matrix with nonzero elements satisfying the property  $a_{ij} \neq 0 \iff (i, j) \in \mathcal{E}$ . The graph is called *weighted* whenever the elements of its adjacency matrix are other than just 0-1 elements. Here, we mostly use weighted graphs with position-dependent adjacency elements. A graph  $G$  is called *undirected* iff the adjacency matrix  $A$  is symmetric (or  $A^T = A$ ). The set of *neighbors* of node  $i$  is defined by

$$N_i = \{j \in \mathcal{V} : a_{ij} \neq 0\} = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}. \quad (1)$$

**Note 1.** Throughout the paper, for simplicity of notation, we assume  $a_{ii} = 0$  for all  $i$  (or the graphs have no loops meaning  $(i, i) \notin \mathcal{E}$ ).

Let  $q_i \in \mathbb{R}^m$  denote the position of node  $i$  for all  $i \in \mathcal{V}$ . The vector  $q = \text{col}(q_1, \dots, q_n) \in Q = \mathbb{R}^{mn}$  is called the *configuration* of all nodes of the graph. A *framework* (or *structure*) is a pair  $(G, q)$  that consists of a graph and the configuration of its nodes.

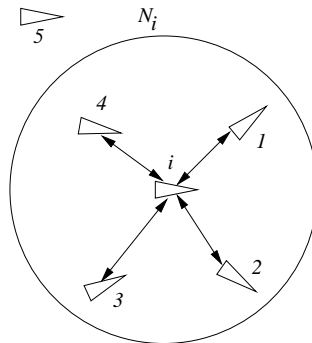


Figure 1: An agent and its neighbors in a spherical neighborhood.

Consider a group of *dynamic agents* (or *particles*) with equation of motion

$$\begin{cases} \dot{q}_i &= p_i, \\ \dot{p}_i &= u_i, \end{cases} \quad (2)$$

where  $q_i, p_i, u_i \in \mathbb{R}^m$  (e.g.  $m = 2, 3$ ) and  $i \in \mathcal{V}$ . The benefit of using particle-based models of flocks compared to continuum models is that one cannot take inter-agent sensing, communication, and computational issues for granted.

Let  $r > 0$  denote the *interaction range* between two agents. An open ball with radius  $r$  (see Fig. 1) determines the set of *spatial neighbors* of agent  $i$  that is denoted by

$$N_i = \{j \in \mathcal{V} : \|q_j - q_i\| < r\} \quad (3)$$

where  $\|\cdot\|$  is the Euclidean norm in  $\mathbb{R}^m$ . Given an interaction range  $r > 0$ , a *spatially induced graph*  $G(q) = (\mathcal{V}, \mathcal{E}(q))$  can be specified by  $\mathcal{V}$  and the set of edges

$$\mathcal{E}(q) = \{(i, j) \in \mathcal{V} \times \mathcal{V} : \|q_j - q_i\| < r, i \neq j\} \quad (4)$$

that clearly depends on  $q$ . We refer to the graph  $G(q)$  as a *net* and to the structure  $(G(q), q)$  as a *framenet*.

The topology of a wireless sensor network with a radio range  $r$  is an example of a net [1]. If the interaction range of all agents is the same, the net  $G(q)$  becomes an undirected graph. A net  $G(q)$  is generically a digraph under either of the following assumptions: 1) the spherical neighborhoods of agents do not have the same radius, or 2) every agent uses a conic neighborhood to determine its neighbors [40, 31]. In this paper, all nets are undirected graphs.

## 2.2 $\alpha$ -Lattices and Quasi $\alpha$ -Lattices

One of our objectives is to design flocking protocols with capabilities that allow a group of dynamic agents to maintain identical inter-agent distances over a net  $G(q)$ . This motivates us to consider the following set of inter-agent algebraic constraints:

$$\|q_j - q_i\| = d, \quad \forall j \in N_i(q) \quad (5)$$

A configuration  $q$  that is the solution of the set of constraints in (5) frequently appears in this paper. Hence, we find it convenient to define these solutions as lattice-type objects.

**Definition 1.** ( $\alpha$ -lattice) An  $\alpha$ -lattice is a configuration  $q$  satisfying the set of constraints in (5). We refer to  $d$  and  $\kappa = r/d$  as the *scale* and *ratio* of an  $\alpha$ -lattice, respectively.

Keep in mind that the net induced by an  $\alpha$ -lattice is not required to be connected. Furthermore, all edges of a structure  $(G(q), q)$  that is induced by an  $\alpha$ -lattice have the same length. Some common examples of 2-D and 3-D  $\alpha$ -lattices (with interaction range  $r = \kappa d$  so that  $\kappa = 1 + \varepsilon$  and  $0 < \varepsilon \ll 1$ ) include *square lattices*, *hexagonal lattices* and the vertex sets of *regular polygons* and *platonic solids*. Fig. 2 illustrates other examples of  $\alpha$ -lattices.

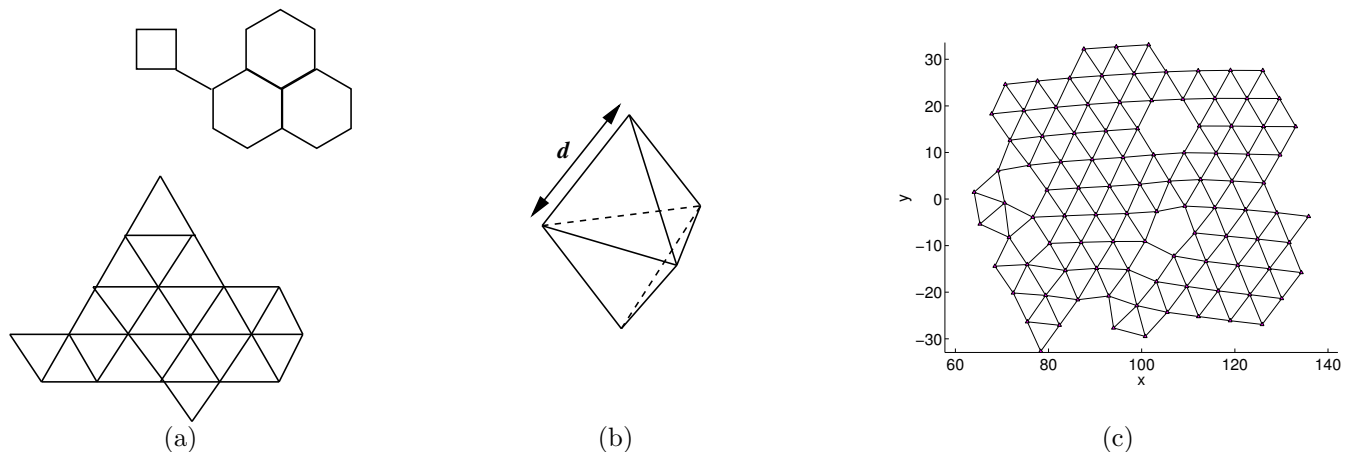


Figure 2: Examples of  $\alpha$ -lattices and quasi  $\alpha$ -lattices: (a) a 2-D  $\alpha$ -lattice, (b) a 3-D  $\alpha$ -lattice, and (c) a 2-D quasi  $\alpha$ -lattice with  $n = 100$  nodes.

We also need to use a slightly deformed versions of  $\alpha$ -lattices that are defined as follows:

**Definition 2.** (quasi  $\alpha$ -lattice) A *quasi  $\alpha$ -lattice* is a configuration  $q$  satisfying the following set of inequality constraints:

$$-\delta \leq \|q_j - q_i\| - d \leq \delta, \quad \forall (i, j) \in \mathcal{E}(q) \quad (6)$$

where  $\delta \ll d$  is the *edge-length uncertainty*,  $d$  is the scale, and  $\kappa = r/d$  is the ratio of the quasi  $\alpha$ -lattice.

Fig. 2 (c) illustrates a quasi  $\alpha$ -lattice that is computed numerically using the second flocking algorithm (or protocol (26)).

### 2.3 Deviation Energy of Conformations

To measure the degree in which a configuration  $q$  differs from an  $\alpha$ -lattice, we use the following *deviation energy*

$$E(q) = \frac{1}{(|\mathcal{E}(q)| + 1)} \sum_{i=1}^n \sum_{j \in N_i} \psi(\|q_j - q_i\| - d) \quad (7)$$

where  $\psi(z) = z^2$  is called a pairwise potential (note that other scalar potentials can be used as well). The deviation energy can be viewed as a nonsmooth potential function for a system of  $n$  particles. Interestingly,  $\alpha$ -lattices are *global minima* of this potential function and achieve the minimum value of zero. For a quasi  $\alpha$ -lattice  $q$  with an edge-length uncertainty of  $\delta$ , the deviation energy is given by

$$E(q) \leq \frac{|\mathcal{E}(q)|}{|\mathcal{E}(q)| + 1} \delta^2 \leq \delta^2 = \varepsilon^2 d^2, \quad \varepsilon \ll 1$$

which means *quasi  $\alpha$ -lattices are low-energy conformations of  $n$  points*. The order of magnitude of the deviation energy of the quasi  $\alpha$ -lattice in Fig. 2 (c) is  $10^{-3}$  (for  $d = 7$  and  $\kappa = 1.2$ ).

### 2.4 $\sigma$ -Norms and Bump Functions

The  $\sigma$ -norm of a vector is a map  $\mathbb{R}^m \rightarrow \mathbb{R}_{\geq 0}$  (not a norm) defined as

$$\|z\|_{\sigma} = \frac{1}{\epsilon} [\sqrt{1 + \epsilon \|z\|^2} - 1] \quad (8)$$

with a parameter  $\epsilon > 0$  and a gradient  $\sigma_{\epsilon}(z) = \nabla \|z\|_{\sigma}$  given by

$$\sigma_{\epsilon}(z) = \frac{z}{\sqrt{1 + \epsilon \|z\|^2}} = \frac{z}{1 + \epsilon \|z\|_{\sigma}} \quad (9)$$

The parameter  $\epsilon$  of the  $\sigma$ -norm remains fixed throughout the paper. One might wonder why do we even need to define a new norm. The map  $\|z\|_{\sigma}$  is differentiable everywhere, but  $\|z\|$  is not differentiable at  $z = 0$ . Later, this property of  $\sigma$ -norms is used for construction of smooth collective potential functions for groups of particles.

*Remark 1.* The function  $\|z\|_{\sigma}$  was effectively used in [32] as a penalty function in construction of potential function for formation control. This led to design and analysis of a distributed algorithm for formation stabilization with bounded control.

*Remark 2.* The map  $\sigma_{\epsilon}(z)$  is a uniformly bounded function satisfying the following properties: i)  $z^T \sigma_{\epsilon}(z) > 0, \forall z \neq 0$ , ii)  $\|\sigma_{\epsilon}(z)\| < 1/\sqrt{\epsilon}$  for all  $z$ , and iii)  $(x - y)^T (\sigma_1(x) - \sigma_1(y)) > 0, \forall x \neq y$ .

A *bump function* is a scalar function  $\rho_h(z)$  that smoothly varies between 0 and 1. Here, we use bump functions for construction of smooth potential functions with finite cut-offs and smooth adjacency matrices. One possible choice is the following bump function introduced in [46]

$$\rho_h(z) = \begin{cases} 1, & z \in [0, h) \\ \frac{1}{2} [1 + \cos(\pi \frac{z-h}{1-h})], & z \in [h, 1] \\ 0 & \text{otherwise.} \end{cases} \quad (10)$$

where  $h \in (0, 1)$ . One can show that  $\rho_h(z)$  is a  $C^1$ -smooth function with the property that  $\rho'_h(z) = 0$  over the interval  $[1, \infty)$  and  $|\rho'_h(z)|$  is uniformly bounded in  $z$ . Using this bump function, we can define a *spatial adjacency matrix*  $A(q)$  via its elements by

$$a_{ij}(q) = \rho_h(\|q_j - q_i\|_{\sigma} / r_{\alpha}) \in [0, 1], \quad j \neq i \quad (11)$$

where  $r_{\alpha} = \|r\|_{\sigma}$  and  $a_{ii}(q) = 0$  for all  $i$  and  $q$ . For  $h = 1$ ,  $\rho_h(z)$  is an indicator function that is equal to 1 over the interval  $[0, 1)$  and 0, otherwise. The use of an indicator bump function leads to a net with 0-1 position-dependent adjacency elements.

## 2.5 Collective Potential Functions

The collective potential function  $V(q)$  of a group of agents is a non-negative function  $V : \mathbb{R}^{mn} \rightarrow \mathbb{R}_{\geq 0}$  with the property that any solution of the set of algebraic constraints in (5) is “closely related to” a local minima of  $V(q)$  and vice versa. The sense in which these two quantities are closely related is measured using the deviation energy that acts as a distance function. In this paper, a collective potential is a smooth version of a deviation energy function with a scalar pairwise potential that has a *finite cut-off*. This means that there exists a finite interaction range  $r > 0$  such that  $\psi'(z) = 0, \forall z \geq r$ . This feature turns out to be the fundamental source of scalability of our flocking algorithms.

*Remark 3.* Generalized Lennard-Jones functions and exponentially vanishing maps do not have finite cut-offs and are inadequate for our purpose without any modifications. A common approach to create a pairwise potential with a finite cut-off is “soft cutting” in which a pairwise potential is multiplied by a bump function (as in [31]). Here, we use an alternative approach by softly cutting action functions and then using their integrals as pairwise potentials. This way the derivative of the bump function never appears in the action function (i.e. derivative of a pairwise potential), and thereby a negative bump in the action function near  $z = r$  is avoided.

Let  $\psi(z) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$  be an *attractive/repulsive* pairwise potential with a global minimum at  $z = d$  and a finite cut-off at  $r$ . Then, the following function

$$\varphi(q) = \frac{1}{2} \sum_i \sum_{j \neq i} \psi(\|q_j - q_i\| - d) \quad (12)$$

is a collective potential that is not differentiable at singular configurations in which two distinct nodes coincide, or  $q_i = q_j$ . To resolve this problem, we use the set of algebraic constraints in (5) that are rewritten in terms of  $\sigma$ -norms as

$$\|q_j - q_i\|_\sigma = d_\alpha, \quad \forall j \in N_i(q) \quad (13)$$

where  $d_\alpha = \|d\|_\sigma$ . These constraints induce a *smooth collective potential function* in the form:

$$V(q) = \frac{1}{2} \sum_i \sum_{j \neq i} \psi_\alpha(\|q_j - q_i\|_\sigma) \quad (14)$$

where  $\psi_\alpha(z)$  is a smooth pairwise attractive/repulsive potential (defined in equation (16)) with a finite cut-off at  $r_\alpha = \|r\|_\sigma$  and a global minimum at  $z = d_\alpha$ .

To construct a smooth pairwise potential with finite cut-off, we integrate an action function  $\phi_\alpha(z)$  that vanishes for all  $z \geq r_\alpha$ . Define this action function as

$$\begin{aligned} \phi_\alpha(z) &= \rho_h(z/r_\alpha)\phi(z - d_\alpha) \\ \phi(z) &= \frac{1}{2}[(a+b)\sigma_1(z+c) + (a-b)] \end{aligned} \quad (15)$$

where  $\sigma_1(z) = z/\sqrt{1+z^2}$  and  $\phi(z)$  is an uneven sigmoidal function with parameters that satisfy  $0 < a \leq b$ ,  $c = |a-b|/\sqrt{4ab}$  to guarantee  $\phi(0) = 0$ . The pairwise *attractive/repulsive potential*  $\psi_\alpha(z)$  in (14) is defined as

$$\psi_\alpha(z) = \int_{d_\alpha}^z \phi_\alpha(s) ds. \quad (16)$$

Functions  $\phi_\alpha(z)$  and  $\psi_\alpha(z)$  are depicted in Fig. 3.

## 2.6 Graph Laplacians and Velocity Matching

Graph Laplacians are very useful in analysis of velocity matching/alignment. Consider a graph  $G$  of order  $n$  with adjacency matrix  $A = [a_{ij}]$ . The *degree matrix* of  $G$  is a diagonal matrix  $\Delta = \Delta(A)$  with diagonal elements  $\sum_{j=1}^n a_{ij}$  that are row-sums of  $A$ . The scalar *graph Laplacian*  $L = [l_{ij}]$  is an  $n \times n$  matrix associated with graph  $G$  that is defined as

$$L = \Delta(A) - A. \quad (17)$$

Laplacian matrix  $L$  always has a right eigenvector of  $\mathbf{1}_n = (1, \dots, 1)^T$  associated with eigenvalue  $\lambda_1 = 0$ . The following lemma summarizes the basic properties of graph Laplacians:



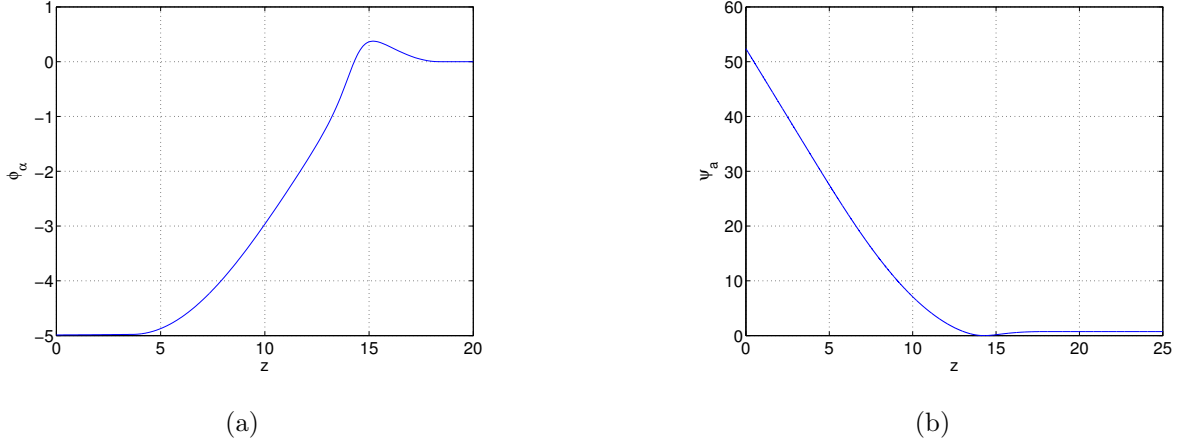


Figure 3: The action and potential functions with finite cut-offs: (a)  $\phi_\alpha(z)$  and (b)  $\psi_\alpha(z)$ .

**Lemma 1.** Let  $G = (\mathcal{V}, \mathcal{E})$  be an undirected graph of order  $n$  with a non-negative adjacency matrix  $A = A^T$ . Then, the following statements hold:

i)  $L$  is a positive semidefinite matrix that satisfies the following sum-of-squares (SOS) property:

$$z^T L z = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}} a_{ij} (z_j - z_i)^2, \quad z \in \mathbb{R}^n; \quad (18)$$

ii) The graph  $G$  has  $c \geq 1$  connected components iff  $\text{rank}(L) = n - c$ . Particularly,  $G$  is connected iff  $\text{rank}(L) = n - 1$ ;

iii) Let  $G$  be a connected graph, then

$$\lambda_2(L) = \min_{z \perp \mathbf{1}_n} \frac{z^T L z}{\|z\|^2} > 0 \quad (19)$$

**Proof.** All three results are well-known in the field of algebraic graph theory and their proofs can be found in Godsil and Royle [12].  $\square$

The quantity  $\lambda_2(L)$  is known as *algebraic connectivity* of a graph [10]. In [33], it was shown that the speed of convergence of a linear consensus protocol (defined in (22)) is equal to  $\lambda_2 > 0$ . This consensus protocol will appear as a *velocity matching* term in all flocking algorithms that will be presented in this paper. Particularly, we use *m-dimensional graph Laplacians*<sup>2</sup> defined by

$$\hat{L} = L \otimes \mathbf{1}_m, \quad (20)$$

where  $\otimes$  denotes the Kronecker product. This multi-dimensional Laplacian satisfies the following SOS property:

$$z^T \hat{L} z = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}} a_{ij} \|z_j - z_i\|^2, \quad z \in \mathbb{R}^{mn} \quad (21)$$

where  $z = \text{col}(z_1, z_2, \dots, z_n)$  and  $z_i \in \mathbb{R}^m$  for all  $i$ . Matrix  $\hat{L}$  can be viewed as the Laplacian of a graph with adjacency matrix  $\hat{A} = A \otimes \mathbf{1}_m$ . There is an important relationship between *m-dimensional Laplacians* and velocity matching in flocks. Let  $G$  be an undirected graph with adjacency matrix  $A = [A_{ij}]$  and suppose each agent applies the following input

$$u_i = \sum_{j \in N_i} a_{ij} (p_j - p_i). \quad (22)$$

This is the multi-variable version of the consensus protocol in [33]. Define a *velocity mismatch function* as  $K = \frac{1}{2} \sum_i \|v_i\|^2$  with  $v_i = p_i - \bar{p}$  and note that  $\bar{p} = \frac{1}{n} \sum_i p_i$  is invariant in time. Then  $\dot{K} = -\frac{1}{2} \sum_{(i,j) \in \mathcal{E}} a_{ij} \|v_j - v_i\|^2 \leq 0$  and  $u_i$  acts as a *damping force*. Under the condition of connectivity of  $G$ , one can show that  $p_i(t) \rightarrow \alpha$  as  $t \rightarrow \infty$  for all  $i$  (see [33] for further details).

<sup>2</sup>In the area of coordination of multi-vehicle systems, *m-dimensional Laplacians* were first used by Fax & Murray [9] for formation stabilization.

### 3 Flocking Algorithms for Free-Space

In this section, we present a set of distributed algorithms for flocking in free-space, or *free-flocking*. (The flocking algorithms with obstacle avoidance capabilities are presented in Section 7.) We refer to a physical agent with equation of motion  $\dot{q}_i = u_i$  as an  $\alpha$ -agent.  $\alpha$ -agents correspond to birds, or members of a flock. This terminology is used to distinguish between physical agents and certain virtual agents that will be introduced later. Informally speaking, an  $\alpha$ -agent has a tendency to stay at a distance  $d > 0$  from all of its neighboring  $\alpha$ -agents (this is the reason behind the name “ $\alpha$ -lattice”). In free-flocking, each  $\alpha$ -agent applies a control input that consists of three terms:

$$u_i = f_i^g + f_i^d + f_i^\gamma \quad (23)$$

where  $f_i^g = -\nabla_{q_i} V(q)$  is a *gradient-based term*,  $f_i^d$  is a *velocity consensus/alignment term* that acts as a damping force, and  $f_i^\gamma$  is a *navigational feedback* due to a *group objective*. Examples of a group objective are moving towards a destination (migration) or flocking out of a crowded space with few exits. We propose two distributed algorithms that can be used for creation of flocking motion in  $\mathbb{R}^m$  (dimensions 1, 2, 3 are of great interest):

**Algorithm 1:**  $u_i = u_i^\alpha$  with

$$u_i^\alpha = \sum_{j \in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma) \mathbf{n}_{ij} + \sum_{j \in N_i} a_{ij}(q)(p_j - p_i) \quad (24)$$

where  $\mathbf{n}_{ij}$  is a vector along the line connecting  $q_i$  to  $q_j$  and is given by

$$\mathbf{n}_{ij} = \sigma_\epsilon(q_j - q_i) = \frac{q_j - q_i}{\sqrt{1 + \epsilon\|q_j - q_i\|^2}} \quad (25)$$

and  $0 < \epsilon < 1$  is a *fixed* parameter of the  $\sigma$ -norm. Algorithm 1 has *no navigational feedback* and is known as the  $(\alpha, \alpha)$  *flocking protocol* [31] because it states the interaction rules between two  $\alpha$ -agents.

**Algorithm 2:**  $u_i = u_i^\alpha + u_i^\gamma$ , or

$$u_i = \sum_{j \in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma) \mathbf{n}_{ij} + \sum_{j \in N_i} a_{ij}(q)(p_j - p_i) + f_i^\gamma(q_i, p_i) \quad (26)$$

where  $u_i^\gamma$  is the *navigational feedback* that is given by

$$u_i^\gamma := f_i^\gamma(q_i, p_i, q_r, p_r) = -c_1(q_i - q_r) - c_2(p_i - p_r), \quad c_1, c_2 > 0. \quad (27)$$

The pair  $(q_r, p_r) \in \mathbb{R}^m \times \mathbb{R}^m$  is the state of a  $\gamma$ -agent. A  $\gamma$ -agent is dynamic/static agent that represents a group objective. Let  $(q_d, p_d)$  be a fixed pair of  $m$ -vectors that denote the initial position and velocity of a  $\gamma$ -agent. A *dynamic  $\gamma$ -agent* has the following model

$$\begin{cases} \dot{q}_r &= p_r, \\ \dot{p}_r &= f_r(q_r, p_r), \end{cases} \quad (28)$$

with  $(q_r(0), p_r(0)) = (q_d, p_d)$ . A *static  $\gamma$ -agent* has a fixed state that is equal to  $(q_d, p_d)$  for all time.

**Note 2.** For those readers who are familiar with the meaning and application of  $\gamma$ -agents in [31],  $\gamma$ -agents in this paper have a completely different meaning and function compared to those described in [31] for obstacle avoidance.

The design of  $f_r(q_r, p_r)$  for a dynamic  $\gamma$ -agent is part of tracking control design for a group of agents. For example, the choice of  $f_r \equiv 0$  leads to a  $\gamma$ -agent that moves along a straight line with a desired velocity  $p_d$ .

During the analysis of protocols (24) and (26), we will see that despite the similarities between certain terms in these protocols, *the collective behavior of a group of agents that use Algorithm 1 is drastically different than a group of agents applying Algorithm 2.*

*Remark 4.* We will demonstrate that *protocol (24) unifies all three rules of Reynolds* in a single equation [30]. This is not the result of applying a gradient-based algorithm. It is due to the fact that the gradient-based term of the protocol can be expressed in terms of position-dependent stress elements of a graph. A detailed discussion of this topic is presented in Section 6.

*Remark 5.* It turns out that protocol (24) leads to flocking behavior only for a *very restricted set of initial states*. For generic set of initial states and large number of agents (e.g.  $n > 10$ ), protocol (24) most likely fails to produce flocking behavior and instead leads to *regular fragmentation* as illustrated in Fig. 4. Fragmentation is a pitfall of flocking. In contrast, protocol (26) never leads to fragmentation. As a result, only Algorithm 2 is truly a flocking algorithm. However, analysis of Algorithm 1 provides a valuable insight into the analysis of Algorithm 2.

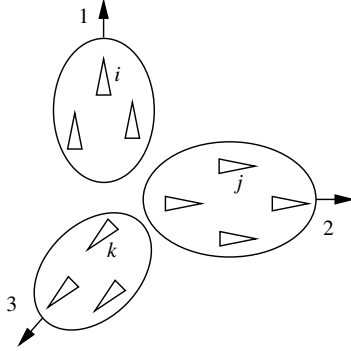


Figure 4: The fragmentation phenomenon.

## 4 Collective Dynamics

The *collective dynamics* of a group of  $\alpha$ -agents applying protocol (26) (or (24)) is in the form:

$$\text{collective dynamics: } \begin{cases} \dot{q} &= p \\ \dot{p} &= -\nabla V(q) - \hat{L}(q)p + f_\gamma(q, p, q_r, p_r) \end{cases} \quad (29)$$

where  $V(q)$  is a smooth collective potential function given in (14) and  $\hat{L}(q)$  is the  $m$ -dimensional Laplacian of the net  $G(q)$  with a position-dependent adjacency matrix  $A(q) = [a_{ij}(q)]$ . Note that for Algorithm 1,  $f_\gamma \equiv 0$ .

The first expected result is that with  $f_\gamma \equiv 0$ , system (29) is a *dissipative particle system* with Hamiltonian

$$H(q, p) = V(q) + \sum_{i=1}^n \|p_i\|^2. \quad (30)$$

This is due to  $\dot{H} = -p^T \hat{L}(q)p \leq 0$  and the fact that the multi-dimensional graph Laplacian  $\hat{L}(q)$  is a positive semidefinite matrix for all  $q$ .

The key in stability analysis of collective dynamics is employing a correct coordinate system that allows the use of LaSalle's invariance principle. The naive approach is to use  $H(q, p)$  in the  $(q, p)$ -coordinates. The reason such an approach does not work (for most cases of interest) is that one cannot establish the boundedness of solutions. During fragmentation, the solution cannot remain bounded. Therefore, we propose the use of a *moving frame* to analyze the stability of flocking motion as suggested in [31].

Consider a moving frame that is centered at  $q_c$ —the center of mass (CM) of all particles. Let  $\text{Ave}(z) = \frac{1}{n} \sum_{i=1}^n z_i$  denote the average of the  $z_i$ 's with  $z = \text{col}(z_1, \dots, z_n)$ . Let  $q_c = \text{Ave}(q)$  and  $p_c = \text{Ave}(p)$  denote the position and velocity of the origin of the moving frame. Then  $\dot{q}_c(t) = p_c(t)$  and  $\dot{p}_c(t) = \text{Ave}(u(t))$ .

Our objective is to separate the analysis of the motion of the center of the group with respect to the reference frame from the collective motion of the particles in the moving frame. The position and velocity of agent  $i$  in the moving frame is given by

$$\begin{cases} x_i &= q_i - q_c, \\ v_i &= p_i - p_c. \end{cases} \quad (31)$$

The relative positions and velocities remain the same in the moving frame, i.e.  $x_j - x_i = q_j - q_i$  and  $v_j - v_i = p_j - p_i$ . Thus,  $V(q) = V(x)$  and  $\nabla V(q) = \nabla V(x)$ . The  $(\alpha, \alpha)$  protocol in the moving frame can be expressed as

$$u_i^\alpha = \sum_{j \in N_i} \phi_\alpha(\|x_j - x_i\|_\sigma) \mathbf{n}_{ij} + \sum_{j \in N_i} a_{ij}(x) (v_j - v_i)$$

with  $a_{ij}(x) = \rho_h(\|x_j - x_i\|_\sigma / r_\alpha)$ . Our first result is a decomposition lemma that is the basis for posing a structural stability problem for “dynamic flocks” (i.e. a dynamic network with a topology that is a connected net and nodes that are particles).

**Lemma 2.** (*decomposition*) *Suppose that the navigational feedback  $f_\gamma(q, p)$  is linear, i.e. there exists a decomposition of  $f_\gamma(q, p)$  in the following form*

$$f_\gamma(q, p, q_r, p_r) = g(x, v) + \mathbf{1}_n \otimes h(q_c, p_c, q_r, p_r). \quad (32)$$

*Then, the collective dynamics of a group of agents applying protocol (26) (or (24)) can be decomposed as  $n$  second-order systems in the moving frame*

$$\text{structural dynamics: } \begin{cases} \dot{x} &= v \\ \dot{v} &= -\nabla V(x) - \hat{L}(x)v + g(x, v) \end{cases} \quad (33)$$

*and one second-order system in the reference frame*

$$\text{translational dynamics: } \begin{cases} \dot{q}_c &= p_c \\ \dot{p}_c &= h(q_c, p_c, q_r, p_r) \end{cases} \quad (34)$$

where

$$\begin{aligned} g(x, v) &= -c_1 x - c_2 v \\ h(q_c, p_c, q_r, p_r) &= -c_1(q_c - q_r) - c_2(p_c - p_r) \end{aligned} \quad (35)$$

and  $(q_r, p_r)$  is the state of the  $\gamma$ -agent.

**Proof.** See Section A.1 in Appendix A. □

A discussion of the effects of using a nonlinear navigational feedback is presented in Appendix B. In this case, it turns out that the structural and translational dynamics are coupled.

## 5 Stability Analysis of Flocking in Free-Space (Main Results)

According to the decomposition lemma, we are now at the position to define *stable flocking motion* as the combination of the following forms of stability properties:

1. Stability of certain equilibria of the structural dynamics.
2. Stability of a desired equilibrium of the translational dynamics.

The challenging part of stability analysis for a flocking algorithm is to establish part 1. Investigation of part 2 is far simpler than part 1. For example, in the case of Algorithm 1,  $\ddot{q}_c = 0$  and thus the center of the group moves along a straight line. Of course, this is not the case for Algorithm 2 where the group objective determines the aggregate behavior of the flock as a whole.

As far as animal behavior is concerned, the translational dynamics of a flock does not necessarily have to possess an equilibrium point that is stable or asymptotically stable. A flock of birds could circle an area over and over or move in a erratic and unpredictable manner. However, from an engineering stand point, an overall control over the collective behavior of a flock is highly desired. In fact, the very reason to perform flocking for UAVs is to steer a group of vehicles from point A to B as a whole. Thus, in robotics or engineering applications, performing the second task becomes very crucial. As a consequence, flocking protocols such as Algorithm 2 that account for the group objective are beneficial for engineering applications.

The significant differences in the effects of Algorithms 1 and 2 are due to the considerable differences in their respective structural dynamics. Given Algorithm 1, one obtains the following structural dynamics:

$$\Sigma_1 : \begin{cases} \dot{x} &= v \\ \dot{v} &= -\nabla V(x) - \hat{L}(x)v \end{cases} \quad (36)$$

with a *positive semidefinite* Laplacian matrix  $\hat{L}(x)$ . In comparison, the structural dynamics of a group of agents applying Algorithm 2 is in the form

$$\Sigma_2 : \begin{cases} \dot{x} &= v \\ \dot{v} &= -\nabla U_\lambda(x) - D(x)v \end{cases} \quad (37)$$

where  $U_\lambda(x)$  is called the *aggregate potential function* and is defined by

$$U_\lambda(x) = V(x) + \lambda J(x). \quad (38)$$

The map  $J(x) = \frac{1}{2} \sum_{i=1}^n \|x_i\|^2$  is the *moment of inertia* of all particles and  $\lambda = c_1 > 0$  is a parameter of the navigational feedback. Moreover, the damping matrix  $D(x) = c_2 I_m + \hat{L}(x)$  is a *positive definite* matrix with  $c_2 > 0$ .

Before presenting the stability analysis of flocking behavior under Algorithms 1 and 2, we need to define the structural Hamiltonians of systems  $\Sigma_1$  and  $\Sigma_2$  as follows:

$$\begin{aligned} H(x, v) &= V(x) + K(v), \\ H_\lambda(x, v) &= U_\lambda(x) + K(v), \end{aligned} \quad (39)$$

where  $K(v) = \frac{1}{2} \sum_i \|v_i\|^2$  is the velocity mismatch function, or the kinetic energy of the particle system in the moving frame. We also need to define what we mean by “cohesion of a group” and “flocks”.

**Definition 3.** (a cohesive group) Let  $(q(\cdot), p(\cdot)) : t \mapsto \mathbb{R}^{mn} \times \mathbb{R}^{mn}$  be the state trajectory of a group of dynamic agents over the time interval  $[t_0, t_f]$ . We say the group is *cohesive* for all  $t \in [t_0, t_f]$  if there exists a ball of radius  $R > 0$  centered at  $q_c(t) = \text{Ave}(q(t))$  that contains all the agents for all time  $t \in [t_0, t_f]$ , i.e.  $\exists R > 0 : \|x(t)\| \leq R, \forall t \in [t_0, t_f]$ . In general, let  $\|x\|_Q = (x^T Q x)^{\frac{1}{2}}$  be the weighted norm of  $x$  with a positive definite matrix  $Q$ . Then, cohesion of a group can be easily generalized to the boundedness of position of all agents in an ellipsoid by  $\|x(t)\|_Q \leq R, \forall t \in [t_0, t_f]$ .

**Note 3.** In this paper, *cohesion* is a property of the trajectory of a group of agents (as in Definition 3) and “cohesion rule” is another common name of the “flock centering” rule of Reynolds. One should not take one for another by mistake.

**Definition 4.** (flocks, quasi-flocks, & dynamic flocks) The configuration  $q$  of a set of points  $\mathcal{V}$  is called a *flock* with interaction range  $r$  if the net  $G(q)$  is connected.  $q$  is called a *quasi-flock* if  $G(q)$  has a giant component (i.e. a connected subgraph with relatively large number of nodes). A group of  $\alpha$ -agents are called a *dynamic flock* over the time interval  $[t_0, t_f]$  if at every moment  $t \in [t_0, t_f]$ , they are a flock.

The aforementioned definitions formalize some of the commonly used terms such as “flocks” and “cohesion of flocks”. Furthermore, according to the following theorem, these notions are essential in explaining why flocking emerges as a form of collective behavior of  $\alpha$ -agents.

**Theorem 1.** Consider a group of  $\alpha$ -agents applying protocol (24) (Algorithm 1) with structural dynamics  $\Sigma_1$  (defined in (36)). Let  $\Omega_c = \{(x, v) : H(x, v) \leq c\}$  be a level-set of the Hamiltonian  $H(x, v)$  of  $\Sigma_1$  such that for any solution starting in  $\Omega_c$ , the group of agents is a cohesive dynamic flock for all  $t \geq 0$ . Then, the following statements hold:

- i) Almost every solution of the structural dynamics converges to an equilibrium  $(x^*, 0)$  with a configuration  $x^*$  that is an  $\alpha$ -lattice.
- ii) The velocity of all agents asymptotically match in the reference frame.
- iii) Given  $c < c^* = \psi_\alpha(0)$ , no inter-agent collisions occur for all  $t \geq 0$ .

**Proof.** Any solution  $(q(t), p(t))$  of the collective dynamics of  $\alpha$ -agents applying protocol (24) is uniquely mapped to a solution  $(x(t), v(t))$  of the structural dynamics  $\Sigma_1$ . We have

$$\dot{H}(x, v) = -v^T \hat{L}(x) v = -\frac{1}{2} \sum_{(i,j) \in \mathcal{E}(x)} a_{ij}(x) \|v_j - v_i\|^2 \leq 0 \quad (40)$$

which means the structural energy  $H(x, v)$  is monotonically decreasing for all  $t \geq 0$ . In addition,  $H(x(t), v(t)) \leq c$  for all  $t \geq 0$  that implies  $\Omega_c$  is an invariant set. This guarantees that the velocity mismatch is upper bounded by  $c$  because of

$$K(v(t)) \leq H(x(t), v(t)) \leq c, \quad \forall t \geq 0.$$

By assumption, for any solution starting in  $\Omega_c$ , the group is cohesive in all time  $t \geq 0$ . Hence, there exists an  $R > 0$  such that  $\|x(t)\| \leq R, \forall t \geq 0$ . The combination of boundedness of velocity mismatch and group cohesion guarantees boundedness of solutions of  $\Sigma_1$  starting in  $\Omega_c$ . This fact is the result of the following inequality:

$$\|(x(t), v(t))\|^2 := \|x(t)\|^2 + \|v(t)\|^2 \leq R^2 + 2c =: C \quad (41)$$

where  $C > 0$  is a constant.

From LaSalle's invariance principle, all the solutions of  $\Sigma_1$  starting in  $\Omega_c$  converge to the largest invariant set in  $E = \{(x, v) \in \Omega_c : \dot{H} = 0\}$ . However, since the group of  $\alpha$ -agents constitutes a dynamic flock for all  $t \geq 0$ ,  $G(q(t))$  is a connected graph for all  $t \geq 0$ . Thus, based on equation (40), we conclude that the velocities of all agents match in the moving frame, or  $v_1 = \dots = v_n$ . But  $\sum_i v_i = 0$ , therefore,  $v_i = 0$  for all  $i$  (or  $v = 0$ ). This means that the velocity of all agents asymptotically match in the reference frame, or  $p_1 = p_2 = \dots = p_n$ , which proves part ii). Moreover, the configuration  $x$  asymptotically converges to a fixed configuration  $x^*$  that is an extrema of  $V(x)$ , i.e.  $\nabla V(x^*) = 0$ .

Since any solution of the system starting at certain equilibria such as local maxima or saddle points remain in those equilibria for all time, not all solutions of the system converge to a local minima. However, anything but a local minima is an unstable equilibria (a blanket assumption). Thus, almost every solution of the system converges to an equilibrium  $(x^*, 0)$  where  $x^*$  is a local minima of  $V(x)$ . According to Theorem 2, every local minima of  $V(x)$  is an  $\alpha$ -lattice. Therefore,  $x^*$  is an  $\alpha$ -lattice and asymptotically all inter-agent distances between neighboring  $\alpha$ -agents become equal to  $d$ . This finishes the proof of parts i) and ii).

We prove part iii) by contradiction. Assume there exists a time  $t = t_1 > 0$  so that two distinct agents  $k, l$  collide, or  $q_k(t_1) = q_l(t_1)$ . For all  $t \geq 0$ , we have

$$\begin{aligned} V(q(t)) &= \frac{1}{2} \sum_i \sum_{j \neq i} \psi_\alpha(\|q_j - q_i\|_\sigma) \\ &= \psi_\alpha(\|q_k(t) - q_l(t)\|_\sigma) + \frac{1}{2} \sum_{i \in \mathcal{V} \setminus \{k, l\}} \sum_{j \in \mathcal{V} \setminus \{i, k, l\}} \psi_\alpha(\|q_j - q_i\|_\sigma) \\ &\geq \psi_\alpha(\|q_k(t) - q_l(t)\|_\sigma). \end{aligned}$$

Hence,  $V(q(t_1)) \geq \psi_\alpha(0) =: c^*$ . But the velocity mismatch is a non-negative quantity and  $\Omega_c$  is an invariant level-set of  $H$ . This gives

$$V(q(t)) \leq H(x(t), v(t)) - K(v(t)) \leq H(x(t), v(t)) \leq c < c^*, \forall t \geq 0$$

which is in contradiction with an earlier inequality  $V(q(t_1)) \geq c^*$ . Therefore, no two agents collide at any time  $t \geq 0$ .  $\square$

The following theorem provides a geometric characterization of the set of local minima of  $V(q)$ . This result is used in the proof of part i) of Theorem 1.

**Theorem 2.** *Every local minima of  $V(q)$  is an  $\alpha$ -lattice and vice versa.*

**Proof.** Define an  $\varepsilon$ -neighborhood of  $q$  as

$$\mathcal{N}_\varepsilon(q) = \{q' \in \mathbb{R}^{mn} : \|q'_i - q_i\| \leq \varepsilon, \forall i \in \mathcal{V}\}, \quad (42)$$

where  $q' = \text{col}(q'_1, q'_2, \dots, q'_n)$ . A configuration  $q^*$  is called a *local minima* of  $V(q)$  if there exists an  $\varepsilon$ -neighborhood  $\mathcal{N}_\varepsilon(q)$  of  $q^*$  such that  $V(q) \geq V(q^*)$  for all  $q \in \mathcal{N}_\varepsilon(q^*)$ . Keep in mind that  $V(q)$  remains invariant under rotation and translation of all  $q_i$ 's, therefore it does not have any *isolated* local minima.

The collective potential function  $V(q)$  can be decomposed into two terms: a graph-induced potential function by the net  $G(q)$  and an additional term that is an integer factor of  $h_0 = \psi_\alpha(r_\alpha)$ , i.e.

$$V(q) = V_G(q) + k(q)h_0 \quad (43)$$

where

$$\begin{aligned} V_G(q) &= \frac{1}{2} \sum_{(i,j) \in \mathcal{E}(q)} \psi_\alpha(\|q_j - q_i\|_\sigma), \\ k(q) &= \frac{1}{2} (n(n-1) - |\mathcal{E}(q)|) \in \mathcal{I} = \{0, 1, 2, \dots, n(n-1)\}. \end{aligned} \quad (44)$$

Notice that  $k(q)$  is an integer that is directly determined by the number of edges of the net  $G(q)$  ( $(i, j)$  and  $(j, i)$  with  $i \neq j$  count as two edges). The higher the size of the net  $G(q)$ , the lower the value of  $k(q)$ . Let  $Q_k$  denote the set of configurations that induce nets with  $|\mathcal{E}(q)| = n(n-1) - 2k$  edges for a fixed specification  $(d, r)$ . Observe that  $\cup_{k=0}^{n(n-1)} Q_k = \mathbb{R}^{mn}$ . For a fixed  $k \in \mathcal{I}$ , we have

$$V(q) = V_G(q) + kh_0 \geq kh_0 \quad (45)$$

for all  $q \in Q_k$ . Any configuration  $q^*$  that achieves the equality in (45) is a local minima of  $V(q)$  and satisfies  $V_G(q^*) = 0$ . On the other hand,  $\psi_\alpha(z)$  takes the value zero only at  $z = d_\alpha$ , hence the configuration  $q^*$  satisfies

$$\|q_j^* - q_i^*\|_\sigma = d_\alpha \implies \|q_j^* - q_i^*\| = d$$

for all  $j \in N_i(q)$ . This means that  $q^*$  is an  $\alpha$ -lattice. The proof of the converse is rather similar and is omitted.  $\square$

The following result provides a global stability analysis of Algorithm 2 that is useful for creation of flocking motion for generic sets of initial conditions. In comparison to Theorem 1, no assumptions regarding group cohesion or connectivity of the net are made in the following theorem:

**Theorem 3.** *Consider a group of  $\alpha$ -agents applying protocol (26) (Algorithm 2) with  $c_1, c_2 > 0$  and structural dynamics  $\Sigma_2$  (defined in (37)). Assume that the initial velocity mismatch  $K(v(0))$  and inertia  $J(x(0))$  are finite. Then, the following statements hold:*

- i) *The group of agents remain cohesive for all  $t \geq 0$ .*
- ii) *Almost every solution of  $\Sigma_2$  asymptotically converges to an equilibrium point  $(x_\lambda^*, 0)$  where  $x_\lambda^*$  is a local minima of  $U_\lambda(x)$ .*
- iii) *The velocity of all agents asymptotically match in the reference frame.*
- iv) *Assume the initial structural energy of the particle system is less than  $(k+1)c^*$  with  $c^* = \psi_\alpha(0)$  and  $k \in \mathbb{Z}_+$ . Then, at most  $k$  distinct pairs of  $\alpha$ -agents could possibly collide ( $k = 0$  guarantees a collision-free motion).*

**Proof.** First, note that the particle system with structural dynamics  $\Sigma_2$  and Hamiltonian  $H_\lambda(x, v) = U_\lambda(x) + K(v)$  is a strictly dissipative particle system in the moving frame because it satisfies

$$\dot{H}_\lambda(x, v) = -v^T(c_2 I_m + \hat{L}(x))v = -c_2(v^T v) - v^T \hat{L}(x)v < 0, \quad \forall v \neq 0.$$

Hence, the structural energy  $H_\lambda(x, v)$  is monotonically decreasing for all  $(x, v)$  and

$$H_\lambda(x(t), v(t)) \leq H_0 := H_\lambda(x(0), v(0)) < \infty.$$

The finiteness of  $H_0 = V(x(0)) + \lambda J(x(0)) + K(v(0))$  follows from the assumption that the collective potential, the inertia, and the velocity mismatch are all initially finite. Thus, for all  $t \geq 0$ , we have

$$U_\lambda(x(t)) \leq H_0, \quad K(v(t)) \leq H_0.$$

But  $U_\lambda(x) = V(x) + \frac{\lambda}{2}x^T x$  with  $\lambda > 0$  and  $V(x) \geq 0$  for all  $x$ , therefore

$$x^T(t)x(t) \leq \frac{2H_0}{\lambda}, \quad \forall t \geq 0.$$

This guarantees the cohesion of the group of  $\alpha$ -agents for all  $t \geq 0$  because the position of all agents remains in a ball of radius  $R = \sqrt{2H_0/\lambda}$  centered at  $q_c$ . This cohesion property together with boundedness of velocity mismatch, or  $K(v(t)) \leq H_0$ , guarantees boundedness of solutions of the structural dynamics  $\Sigma_2$ . To see this, let  $z = \text{col}(x, v)$ , then

$$\|z(t)\|^2 = x^T(t)x(t) + v^T(t)v(t) \leq 2\left(\frac{1}{\lambda} + 1\right)H_0 =: C(\lambda) < \infty.$$

Part ii) follows from LaSalle's invariance principle. Keep in mind that  $\dot{H}_\lambda(x, v) = 0$  implies  $v = 0$ . Thus, similar to the argument in the proof of Theorem 1, almost every solution of the particle system asymptotically converges to an equilibrium point  $z_\lambda^* = (x_\lambda^*, 0)$  where  $x_\lambda^*$  is a local minima of the aggregate potential function  $U_\lambda(x)$ .

Part iii) follows from the fact that  $v$  asymptotically vanishes. Thus, the velocities of all agents asymptotically match in the reference frame.

To prove part iv), suppose  $H_0 < (k+1)c^*$  and there are more than  $k$  distinct pairs of agents that collide at a given time  $t_1 \geq 0$ . Hence, there must be at least  $k+1$  distinct pairs of agents that collide at time  $t_1$ . This implies the collective potential of the particle system at time  $t = t_1$  is at least  $(k+1)\psi_\alpha(0)$ . However, we have

$$H_0 = V(x(0)) + \lambda J(x(0)) + K(v(0)) \geq V(x(0)) \geq (k+1)\psi_\alpha(0).$$

This contradicts the assumption that  $H_0 < (k+1)c^*$ . Hence, no more than  $k$  distinct pairs of agents can possibly collide at any time  $t \geq 0$ . Finally, with  $k = 0$ , no two agents ever collide.  $\square$

Theorem 3 establishes some critical properties of collective behavior of a group of agents applying Algorithm 2 including cohesion, convergence, asymptotic velocity matching, and collision-avoidance. But, unless one provides a geometric characterization of local minima of  $U_\lambda(q)$  for relatively small  $\lambda > 0$ , it is not possible to establish that the limiting conformation  $x_\lambda^*$  is “closely related” to an  $\alpha$ -lattice. I would like to pose two conjectures that establish this close relationship between geometric and graph theoretic properties of any local minima of  $U_\lambda(q)$  and features of flocks:

**Conjecture 1.** (*connectivity*) Any local minima  $q_\lambda^*$  of  $U_\lambda(q)$  for  $\lambda > 0$  is a flock with interaction range  $r$ .

Conjecture 1 can be equivalently stated as follows: Let  $U_\lambda(q) = V(q) + \lambda J(x)$  be an aggregate potential function, then any local minima of  $U_\lambda(q)$  with  $\lambda > 0$  induces a connected net  $G(q_\lambda^*)$ . The next conjecture states geometric properties of  $q_\lambda^*$ .

**Conjecture 2.** (*quasi  $\alpha$ -lattice*) Let  $U_\lambda(q) = V(q) + \lambda J(x)$  be an aggregate potential function with parameter  $\lambda > 0$ . For any fixed  $n, d, r$  satisfying  $r/d = 1 + \varepsilon$  ( $\varepsilon \ll 1$ ), there exists a  $\lambda^* \ll 1$  so that 1) any local minima  $q_\lambda^*$  of  $U_\lambda(q)$  with  $\lambda \in (0, \lambda^*)$  is a quasi  $\alpha$ -lattice with ratio  $\kappa = 1 + \varepsilon$  and 2)  $q_\lambda^*$  induces a planar graph  $(G(q_\lambda^*), q_\lambda^*)$  in dimensions  $m = 1, 2, 3$ .

Proof/disproof of both conjectures is the subject of ongoing research. Now, we are ready to present a more enhanced version of Theorem 3 with both geometric and graph theoretic relations to flocking.

**Theorem 4.** Consider a group of  $\alpha$ -agents applying protocol (26) (Algorithm 2) with  $c_1, c_2 > 0$  and structural dynamics  $\Sigma_2$ . Assume the initial structural energy is finite and the interaction range satisfies  $r/d = 1 + \varepsilon$  ( $\varepsilon \ll 1$ ). If conjectures 1 and 2 hold, then there exists a  $\lambda^* \ll 1$  such that with  $\lambda \in (0, \lambda^*)$ , almost every solution of  $\Sigma_2$  converges to an equilibrium point  $(x_\lambda^*, 0)$  where  $x_\lambda^*$  is a flock with low deviation energy.

**Proof.** The proof follows from part ii) of Theorem 3 and Conjectures 1 and 2.  $\square$

In simulation results of flocking (see Section 8), the author observed that during flocking, the framements induced by the trajectory of  $\alpha$ -agents were planar graphs<sup>3</sup>. The following theorem, analytically establishes this planarity property as well as a bound on the computational complexity of nets induced by  $\alpha$ -lattices.

**Theorem 5.** Let  $q$  be an  $\alpha$ -lattice of scale  $d > 0$  and ratio  $\kappa > 1$  with  $n$  nodes at distinct positions. Then

- i) The structure  $(G(q), q)$  induced by  $q$  is a planar graph in dimensions  $m = 2, 3$ .
- ii) The net  $G(q)$  has at most  $3n - 6$  links in dimension 2.
- iii) The net  $G(q)$  induced by an  $\alpha$ -lattice with  $n > m+1$  nodes cannot be a complete graph in dimensions  $m = 1, 2, 3$ .

**Proof.** See Section A.2 in Appendix A.  $\square$

Based on part iii) of Theorem 5, an interesting property of  $\alpha$ -lattices is that their spatially induced graphs are never complete graphs if the lattice has a medium to large number of nodes ( $n \geq 4$  nodes in dimension 2). This fact discourages the use of all-to-all networks (or centralized algorithms) for creation of flocking.

*Remark 6.* Part ii) of Theorem 5 states that the total number of the edges of a 2-D flock that is an  $\alpha$ -lattice is linear in  $n$ . This is in agreement with our policy of lowering the communication and computational cost in multi-vehicle systems by using scalable algorithms.

<sup>3</sup>A planar graph is a structure in which no two edges intersect each other anywhere other than their endpoints.



*Remark 7.* According to Theorem 5, the planarity of graphs induced by  $\alpha$ -lattices imposes a restriction on maximum ratio of the interaction range  $r$  to desired distance  $d > 0$ . For example, a cubic lattice  $q$  in dimension  $m$  is a valid  $\alpha$ -lattice with ratio  $\kappa = 1 + \varepsilon$  if  $0 < \varepsilon < \sqrt{m}$ . Otherwise, two nodes that are on opposite sides of the diagonal of a hypercube become neighbors and this invalidates  $q$  as an  $\alpha$ -lattice. Since each node can no longer be equally distanced from all of its neighbors. This is schematically shown in Figure 5 for dimension  $m = 2$ .



Figure 5: The effect of ratio  $\kappa$  on validity of a square lattice as an  $\alpha$ -lattice: (a)  $\kappa = 1 + \varepsilon < \sqrt{2}$  (valid) and (b)  $\kappa = \sqrt{2} + \varepsilon$  (invalid).

## 6 Algorithm 1 vs. Reynolds Rules

In this section, we demonstrate that Algorithm 1 (or the  $(\alpha, \alpha)$  protocol in (24)) embodies extended forms of all three rules of Reynolds in a single equation [30]. In [40], the flock centering rule of Reynolds is further clarified by saying that “each agent moves towards the center of its neighboring agents”. The term “center” means center of mass (CM). This does not clarify when rules 1 and 2 of Reynolds need to be applied at a given situation. This issue is resolved during our attempt to extract Reynolds rules from Algorithm 1.

The key tool in our analysis are *stress elements of a graph* [46]. Let us define the stress elements associated with edge  $(i, j)$  of the net  $G(q)$  as

$$s_{ij}(q) = \frac{\phi_\alpha(\|q_j - q_i\|_\sigma)}{1 + \epsilon\|q_j - q_i\|_\sigma}, \quad (i, j) \in \mathcal{E}(q). \quad (46)$$

The stress elements between non-neighboring agents are defined to be zero. The  $(\alpha, \alpha)$  flocking protocol can be expressed in terms of stress and adjacency elements as follows:

$$u_i^\alpha = \sum_{j \in N_i(q)} s_{ij}(q)(q_j - q_i) + \sum_{j \in N_i(q)} a_{ij}(q)(p_j - p_i) = u_i^g + u_i^{vm}. \quad (47)$$

From the above equation, all three rules of Reynolds follow. The second term represents the velocity matching rule (or rule 3) and the first term embodies both the flock centering and separation rules (rules 1 & 2).

To demonstrate this fact, we consider two cases: i)  $S_i(q) \neq 0$  and ii)  $S_i(q) = 0$  where  $S_i(q) = \sum_{j \in N_i} s_{ij}(q)$  turns out to be the “quantitative measure” that determines whether rule 1 or rule 2 need to be applied at a given state. In case i), we have

$$u_i^g = \sum_{j \in N_i} s_{ij}(q)(q_j - q_i) = S_i(q)(q_i - \langle q_i \rangle) \quad (48)$$

where  $\langle q_i \rangle$  is the weighted average of the position of the neighbors of agent  $i$ , i.e.

$$\langle q_i \rangle = \frac{\sum_{j \in N_i} s_{ij}(q)q_j}{\sum_{j \in N_i} s_{ij}(q)}. \quad (49)$$

Clearly, the *weights are stress elements* that depend on the relative position of neighboring agents. The weighted center  $\langle q_i \rangle$  is undefined for  $S_i(q) = 0$ . This is why we need to analyze case ii).

Before discussing case ii), let us mention that each agent follows the following rules:

$$\text{action } u_i^g \text{ of agent } i = \begin{cases} \text{move towards the weighted center of the neighbors,} & S_i(q) > 0 \\ \text{move away from the weighted center of the neighbors,} & S_i(q) < 0 \end{cases} \quad (50)$$

Here, “weighted center” means  $\langle q_i \rangle$  (defined in (49)). As a byproduct, we obtain a clear guideline that allows us to determine whether the “separation rule” or the “cohesion rule” is applicable in state  $(q, p)$ .

It remains to analyze case ii) with  $S_i(q) = 0$ . Let us define the sets of *friends* of agent  $i$ . Define the set of *positive*, *negative*, and *neutral friends* of agent  $i$ , respectively, as

$$F_i^+ = \{j \in N_i(q) : s_{ij}(q) > 0\}, F_i^- = \{j \in N_i(q) : s_{ij}(q) < 0\}, F_i^0 = \{j \in N_i(q) : s_{ij}(q) = 0\}. \quad (51)$$

Clearly,  $\{F_i^+(q), F_i^-(q), F_i^0(q)\}$  is a partition of  $N_i(q)$  and  $F_i^0(q)$  is always a nonempty set because  $i \in F_i^0$  (i.e. each agent is its own neutral friend). Let us define the sums of stress elements between agent  $i$  and its positive and negative friends as

$$S_i^+(q) = \sum_{j \in F_i^+(q)} s_{ij}(q)q_j, S_i^-(q) = \sum_{j \in F_i^-(q)} s_{ij}(q)q_j.$$

We have  $S_i(q) = S_i^+(q) + S_i^-(q) = 0$ , thus

$$\begin{aligned} u_i^g &= \sum_{j \in N_i(q)} s_{ij}(q)q_j - S_i(q)q_i \\ &= \sum_{j \in F_i^+} s_{ij}(q)q_j + \sum_{j \in F_i^-} s_{ij}(q)q_j \\ &= S_i^+(q)\langle q_i \rangle^+ + S_i^-(q)\langle q_i \rangle^- \end{aligned}$$

or

$$u_i^g = S_i^+(q) (\langle q_i \rangle^+ - \langle q_i \rangle^-) \quad (52)$$

where  $\langle q_i \rangle^+$  and  $\langle q_i \rangle^-$  denote the *positive and negative weighted centers* of neighbors of agent  $i$  defined by

$$\langle q_i \rangle^+ = \frac{\sum_{j \in F_i^+(q)} s_{ij}(q)q_j}{\sum_{j \in F_i^+(q)} s_{ij}(q)}, \langle q_i \rangle^- = \frac{\sum_{j \in F_i^-(q)} s_{ij}(q)q_j}{\sum_{j \in F_i^-(q)} s_{ij}(q)}. \quad (53)$$

According to (52), given an agent with  $S_i(q) = 0$ , a force parallel to the line connecting the weighted centers of negative and positive friends of agent  $i$  is applied to the agent. This extra rule can be informally stated as “*stay away from the center of your negative friends and move towards the center of your positive friends*”. We call this the *signed friendship* rule. Needless to mention that this rule does not follow from any rules of Reynolds.

At any node of an  $\alpha$ -lattice  $S_i(q) = 0$  and all friends of an  $\alpha$ -agent are neutral friends. According to Theorem 1, *all agents applying the  $(\alpha, \alpha)$  protocol asymptotically become neutral friends of their neighbors* (modulo technical conditions) and form a stress-free structure.

## 7 Flocking with Obstacle Avoidance

In this section, we present a distributed flocking algorithm with multiple obstacle-avoidance capability. The main idea is to use agent-based representation of all nearby (active) obstacles by creating a new species of agents called  $\beta$ -agents. A  $\beta$ -agent is a kinematic agent that is induced by an  $\alpha$ -agent whenever the  $\alpha$ -agent is in close proximity of an obstacle. In the following, we formally define the notion of a  $\beta$ -agent and specify the interaction protocol between an  $\alpha$ -agent and a  $\beta$ -agent.

We restrict our study to obstacles that are *connected convex regions* in  $\mathbb{R}^m$  with boundaries that are smooth manifolds. Specifically, we focus on obstacles that are either spheres or infinite walls as shown in Fig. 6. Our approach to obstacle avoidance can be summarized in the following steps:

- 1) Determine the indices  $N_i^\beta$  (to be defined) of the set of obstacles  $O_k$  that are neighbors of  $\alpha$ -agent  $i$ .

- 2) Create a (virtual) kinematic  $\beta$ -agent at  $\hat{q}_{i,k}$  on the boundary of a neighboring obstacle  $O_k$  by projection where  $\hat{q}_{i,k}$  satisfies

$$\hat{q}_{i,k} = \operatorname{argmin}_{x \in O_k} \|x - q_i\| \quad (54)$$

and  $O_k$  is either a closed ball or a closed half space on one side of a hyperplane.

- 3) Add a term  $\psi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)$  to the potential function of a group of  $\alpha$ -agents corresponding to each  $\beta$ -agent at  $\hat{q}_{i,k}$  ( $\psi_\beta(z)$  to be defined).

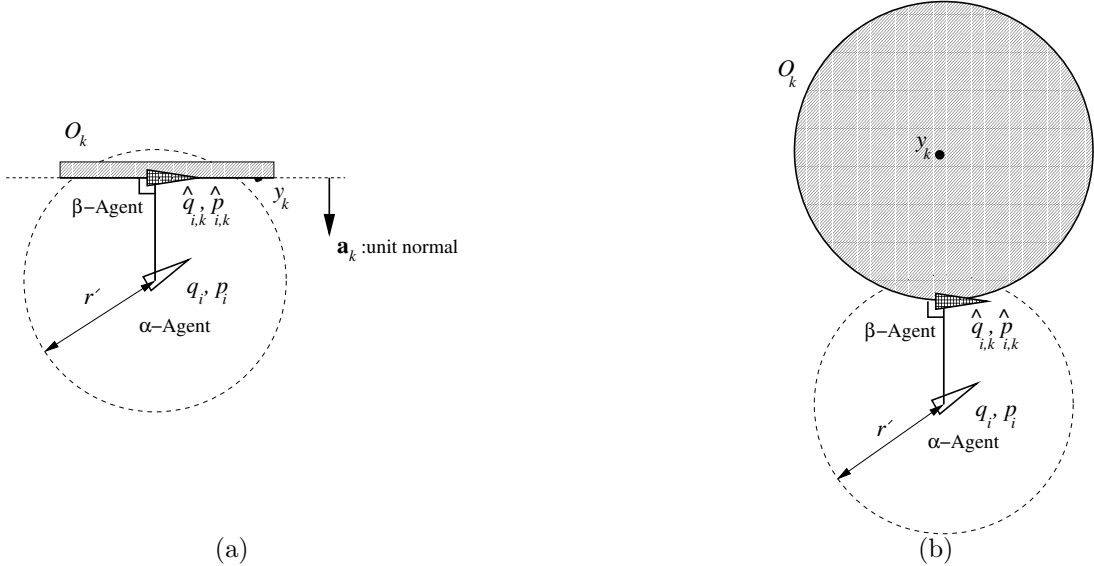


Figure 6: Agent-based representation of obstacles: (a) a wall and (b) a spherical obstacle.

This approach is partially motivated by the work of Khatib [18] and Helbing *et al.* [14]. Fig. 6 schematically illustrates the position of a  $\beta$ -agent induced by an  $\alpha$ -agent in proximity of two types of obstacles. Later, we calculate the velocity of a  $\beta$ -agent based on the velocity of its generating  $\alpha$ -agent. It is not difficult to imagine that notions such as nets, framenets, and  $\alpha$ -lattices can be generalized to  $(\alpha, \beta)$ -nets,  $(\alpha, \beta)$ -framenets, and constrained  $\alpha$ -lattices. We briefly present this generalization and use it for derivation of the flocking algorithm with obstacle-avoidance capability.

### 7.1 $\beta$ -Neighbors of $\alpha$ -Agents and $(\alpha, \beta)$ -Nets

Let  $\mathcal{V}_\alpha = \{1, 2, \dots, n\}$  and  $\mathcal{V}_\beta = \{1', 2', \dots, l'\}$  denote the set of indices of  $\alpha$ -agents and obstacles (or  $\beta$ -agents), respectively. Notice that the prime in elements of  $\mathcal{V}_\beta$  is used to guarantee that  $\mathcal{V}_\alpha \cap \mathcal{V}_\beta = \emptyset$ . An  $\alpha$ -agent is called a *neighbor of an obstacle*  $O_k$  ( $k \in \mathcal{V}_\beta$ ) if and only if the ball  $B_{r'}(q_i)$  and  $O_k$  overlap (as shown in Fig. 6). This form of neighborhood between an  $\alpha$ -agent and an obstacle is a mutual property. Moreover, an  $\alpha$ -agent could possibly have multiple neighboring obstacles. Particularly, this occurs when a group of agents intend to pass through a narrow pathway so that an agent might come within close proximity of multiple obstacles (this will become clear during the squeezing maneuver).

We define the sets of  $\alpha$ -neighbors and  $\beta$ -neighbors of an  $\alpha$ -agent  $i \in \mathcal{V}_\alpha$  as follows:

$$N_i^\alpha = \{j \in \mathcal{V}_\alpha : \|q_j - q_i\| < r\} \quad (55)$$

$$N_i^\beta = \{k \in \mathcal{V}_\beta : \|\hat{q}_{i,k} - q_i\| < r'\} \quad (56)$$

where  $r, r' > 0$  are *interaction ranges* of an  $\alpha$ -agent with neighboring  $\alpha$ -agents and  $\beta$ -agents, respectively. Here, we choose  $r' < r$ , but in general,  $r$  and  $r'$  can be chosen independently.

*Remark 8.* Apparently,  $N_i^\alpha = N_i$ . However, redefining the set of neighbors of an  $\alpha$ -agent is beneficial for the sake of clarity of notations due to the existence of multiple species of agents in our upcoming discussions.

The sets of  $\alpha$ - and  $\beta$ -neighbors of an  $\alpha$ -agent  $i \in \mathcal{V}_\alpha$  naturally define an  $(\alpha, \beta)$ -net that is a spatially induced graph in the form

$$G_{\alpha, \beta}(q) = G_\alpha(q) + G_\beta(q) \quad (57)$$

where  $G_\alpha(q) = (\mathcal{V}_\alpha, \mathcal{E}_\alpha(q))$  is a net induced by configuration of all  $\alpha$ -agents and  $G_\beta(q) = (\mathcal{V}_\beta, \mathcal{E}_\beta(q))$  is a *directed bipartite graph* induced by  $q$  and the set of obstacles  $\mathcal{O} = \{O_k : k \in \mathcal{V}_\beta\}$  where  $\mathcal{E}_\beta(q) \subseteq \mathcal{V}_\alpha \times \mathcal{V}_\beta$ . The condition  $\mathcal{V}_\alpha \cap \mathcal{V}_\beta = \emptyset$  guarantees well-posedness of the definition of the bipartite graph  $G_\beta(q)$ . More explicitly, we have

$$\begin{aligned} \mathcal{E}_\alpha(q) &= \{(i, j) : i \in \mathcal{V}_\alpha, j \in N_i^\alpha\}, \\ \mathcal{E}_\beta(q) &= \{(i, k) : i \in \mathcal{V}_\alpha, k \in N_i^\beta\}, \end{aligned} \quad (58)$$

and  $G_{\alpha, \beta}(q) = (\mathcal{V}_\alpha \cup \mathcal{V}_\beta, \mathcal{E}_\alpha(q) \cup \mathcal{E}_\beta(q))$ . Similarly, an  $(\alpha, \beta)$ -*framenet* is a structure  $(G_{\alpha, \beta}(q), q, \hat{q})$  where  $\hat{q}$  denotes the configuration of all  $\beta$ -agents. Keep in mind that there are no edges between two  $\beta$ -agents.

## 7.2 Constrained $\alpha$ -Lattices

The new set of inter-agent and agent-to-obstacle algebraic constraints for an  $\alpha$ -agent can be specified as follows:

$$\begin{cases} \|q_j - q_i\| &= d, \forall j \in N_i^\alpha \\ \|\hat{q}_{i,k} - q_i\| &= d', \forall k \in N_i^\beta \end{cases} \quad (59)$$

A *constrained  $\alpha$ -lattice* denoted by  $(q, \mathcal{O})$  consists of an  $\alpha$ -lattice  $q$  and a set of obstacles  $\mathcal{O}$  that satisfy the set of constraints in (59). The relevant ratios of a constrained  $\alpha$ -lattice are  $\kappa = r/d$  and  $\kappa' = d'/d = r'/r$  (we assume  $\kappa' = \kappa$ ).

## 7.3 Multi-Species Collective Potentials

To achieve flocking in presence of obstacles, we use the following *multi-species collective potential function* for the particle system:

$$V(q) = c_1^\alpha V_\alpha(q) + c_1^\beta V_\beta(q) + c_1^\gamma V_\gamma(q) \quad (60)$$

where the  $c_1^\alpha, c_1^\beta, c_1^\gamma$  are positive constants and  $(\alpha, \alpha)$ ,  $(\alpha, \beta)$ ,  $(\alpha, \gamma)$  interaction potentials are defined as follows

$$V_\alpha(q) = \sum_{i \in \mathcal{V}_\alpha} \sum_{j \in \mathcal{V}_\alpha \setminus \{i\}} \psi_\alpha(\|q_j - q_i\|_\sigma), \quad (61)$$

$$V_\beta(q) = \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} \psi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma), \quad (62)$$

$$V_\gamma(q) = \sum_{i \in \mathcal{V}_\alpha} (\sqrt{1 + \|q_i - q_r\|^2} - 1). \quad (63)$$

The function  $V_\gamma(q)$  has to do with the navigational objective of a group of  $\alpha$ -agents. The *heterogeneous adjacency* between an  $\alpha$ -agent at  $q_i$  and its neighboring  $\beta$ -agent at  $\hat{q}_{i,k}$  is defined as

$$b_{i,k}(q) = \rho_h(\|\hat{q}_{i,k} - q_i\|_\sigma / d_\beta), \quad (64)$$

where  $d_\beta < r_\beta$  with  $d_\beta = \|d'\|_\sigma, r_\beta = \|r'\|_\sigma$ . Define the *repulsive action function*

$$\phi_\beta(z) = \rho_h(z/d_\beta)(\sigma_1(z - d_\beta) - 1) \quad (65)$$

with  $\sigma_1(z) = z/\sqrt{1+z^2}$ . Notice that  $\phi_\beta(z)$  vanishes smoothly to zero at  $z = d_\beta$  and remains zero for all  $z \geq d_\beta$ . This naturally defines a *repulsive pairwise potential*  $\psi_\beta(z)$  in the form

$$\psi_\beta(z) = \int_{d_\beta}^z \phi_\beta(s) ds \geq 0. \quad (66)$$

Since vehicles/robots/animals in real-life cannot apply unbounded forces, we avoid the use of repulsive potential functions with unbounded derivatives such as  $1/z$  or  $\log(z)$  that are well-known examples of barrier functions. Clearly,  $-2 < \psi'_\beta(z) \leq 0$  for all  $z \in \mathbb{R}$ , and thereby the derivative of  $\psi_\beta(z)$  is uniformly bounded.

## 7.4 Flocking Algorithm in Presence of Obstacles

We are ready to present our main flocking algorithm with obstacle-avoidance capability:

**Algorithm 3:** This algorithm consists of three terms:

$$u_i = u_i^\alpha + u_i^\beta + u_i^\gamma \quad (67)$$

where  $u_i^\alpha$  denotes the  $(\alpha, \alpha)$  interaction terms (i.e. algorithm (24)),  $u_i^\beta$  denotes the  $(\alpha, \beta)$  interaction terms, and  $u_i^\gamma$  is a distributed navigational feedback. We explicitly specify each term in equation (67) as follows:

$$u_i^\alpha = c_1^\alpha \sum_{j \in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma) \mathbf{n}_{i,j} + c_2^\alpha \sum_{j \in N_i^\alpha} a_{ij}(q)(p_j - p_i) \quad (68)$$

$$u_i^\beta = c_1^\beta \sum_{k \in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma) \hat{\mathbf{n}}_{i,k} + c_2^\beta \sum_{j \in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \quad (69)$$

$$u_i^\gamma = -c_1^\gamma \sigma_1(q_i - q_r) - c_2^\gamma (p_i - p_r) \quad (70)$$

where  $\sigma_1(z) = z/\sqrt{1+\|z\|^2}$  and  $c_\eta^\nu$  are positive constants for all  $\eta = 1, 2$  and  $\nu = \alpha, \beta, \gamma$ . The pair  $(q_r, p_r)$  is the state of a static/dynamic  $\gamma$ -agent. The vectors  $\mathbf{n}_{i,j}$  and  $\hat{\mathbf{n}}_{i,k}$  are given by

$$\mathbf{n}_{ij} = \frac{q_j - q_i}{\sqrt{1 + \epsilon\|q_j - q_i\|^2}}, \quad \hat{\mathbf{n}}_{i,k} = \frac{\hat{q}_{i,k} - q_i}{\sqrt{1 + \epsilon\|\hat{q}_{i,k} - q_i\|^2}}.$$

The only missing piece of the puzzle is the method of calculation of position and velocity of  $\beta$ -agents that is discussed in details in the next section.

In terms of sensing requirements, we assume that every  $\alpha$ -agent is equipped with *range sensors* that allows the agent to measure the relative position between the closest point on an obstacle and itself. Both *radars* and *laser radars* (or *ladars*) can be used as range sensors, therefore this assumption is feasible in practice. For the purpose of simulation of flocking behavior, we use the projection of the position of an  $\alpha$ -agent on the boundary of an obstacle.

## 7.5 Calculation of Position and Velocity of $\beta$ -Agents

Given an obstacle  $O_k$  and its neighboring  $\alpha$ -agent with state  $(q_i, p_i)$ , the position and velocity of a  $\beta$ -agent on a wall or a sphere is given by the following lemma:

**Lemma 3.** *Let  $\hat{q}_{i,k}, \hat{p}_{i,k}$  with  $(i, k) \in \mathcal{V}_\alpha \times \mathcal{V}_\beta$  denote the position and velocity of a  $\beta$ -agent generated by an  $\alpha$ -agent with state  $(q_i, p_i)$  on an obstacle  $O_k$ . Then*

- i) *For an obstacle with a hyperplane boundary that has a unit normal  $\mathbf{a}_k$  and passes through the point  $y_k$ , the position and velocity of the  $\beta$ -agent are determined by*

$$\hat{q}_{i,k} = Pq_i + (I - P)y_k, \quad \hat{p}_{i,k} = Pp_i$$

where  $P = I - \mathbf{a}_k \mathbf{a}_k^T$  is a projection matrix.

- ii) *For a spherical obstacle with radius  $R_k$  centered at  $y_k$ , the position and velocity of the  $\beta$ -agent are given by*

$$\hat{q}_{i,k} = \mu q_i + (1 - \mu)y_k, \quad \hat{p}_{i,k} = \mu Pp_i$$

where  $\mu = R_k/\|q_i - y_k\|$ ,  $\mathbf{a}_k = (q_i - y_k)/\|q_i - y_k\|$ , and  $P = I - \mathbf{a}_k \mathbf{a}_k^T$ .

**Proof.** We provide the proof for two cases: a wall and a sphere.

- i) *Wall:* In this case, the boundary of the obstacle is a hyperplane that can be specified by a point  $y_k$  and unit normal vector  $\mathbf{a}_k$  as  $h_k : \mathbf{a}_k^T(y - y_k) = 0$ .

The position of a  $\beta$ -agent on a hyperplane  $\mathbf{a}_k^T(y - y_k) = 0$  is a point  $\hat{q}_{i,k}$  given by

$$\hat{q}_{i,k} = Pq_i + (I - P)y_k \quad (71)$$

One can readily verify that this point is on the hyperplane  $h_k$  and  $q_i - \hat{q}_{i,k}$  is orthogonal to the surface of the hyperplane. Since  $\mathbf{a}_k, y_k$  are fixed, the velocity  $\hat{p}_{i,k} := \dot{\hat{q}}_{i,k}$  of a  $\beta$ -agent on  $h_k$  is given by

$$\hat{p}_{i,k} = Pp_i = p_i - (\mathbf{a}_k^T p_i) \mathbf{a}_k \quad (72)$$

Clearly, the velocity of a  $\beta$ -agent is parallel to the wall, or  $\mathbf{a}_k^T \hat{p}_{i,k} = 0$ . This finishes the proof of part i).

ii) *Spherical Obstacle*: In this case, the boundary of the obstacle is a sphere of radius  $R_k$  centered at  $y_k$ . Define  $\mu$  as

$$\mu = \frac{R_k}{\|q_i - y_k\|} = \frac{R_k}{R_k + \|q_i - \hat{q}_{i,k}\|} \quad (73)$$

and notice that the projection of  $q_i$  on the boundary of the obstacle  $O_k$  can be obtained by a convex combination of  $q_i$  and  $y_k$  as

$$\hat{q}_{i,k} = \mu q_i + (1 - \mu) y_k \quad (74)$$

with  $\mu \in (0, 1]$  (simply because these three points are collinear). The value  $\mu = 1$  corresponds to two cases: 1) as  $R_k \rightarrow \infty$ , a sphere asymptotically approaches a flat surface (or hyperplane) and 2) when an  $\alpha$ -agent and its corresponding  $\beta$ -agent coincide. By differentiating  $\hat{q}_{i,k}$  in time, we get

$$\hat{p}_{i,k} = \dot{\hat{q}}_{i,k} = \mu p_i + \dot{\mu}(q_i - y_k) = \mu p_i + \dot{\mu} \|q_i - y_k\| \mathbf{a}_k$$

where  $\mathbf{a}_k = (q_i - y_k) / \|q_i - y_k\|$  is the unit normal to the sphere at  $\hat{q}_{i,k}$ . Due to the fact that  $y_k, R_k$  are fixed, according to equation (73), we have

$$\dot{\mu} \|q_i - y_k\| + \mu \mathbf{a}_k^T p_i = 0.$$

Hence

$$\hat{p}_{i,k} = \mu p_i - \mu (\mathbf{a}_k^T p_i) \mathbf{a}_k = \mu (I - \mathbf{a}_k \mathbf{a}_k^T) p_i = (\mu P) p_i$$

and this finishes the proof of part ii).  $\square$

The following lemma demonstrates that the second term in  $u_i^\beta$  is in fact a valid damping force. This fact is used later to establish that the overall particle system is dissipative.

**Lemma 4.** *The force  $\hat{f}_d$  between  $\alpha$ -agents and  $\beta$ -agents with elements  $\hat{f}_i^d = \sum_{k \in N_i^\beta} b_{i,k} (p_i - \hat{p}_{i,k})$  is a valid damping force, i.e. let  $K_r = \frac{1}{2} \sum_i \|p_i\|^2$  and suppose  $\dot{p}_i = \hat{f}_i^d$ , then  $\dot{K}_r \leq 0$ .*

**Proof.** By differentiating  $K_r$  in time, we get

$$\dot{K}_r = \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} b_{i,k} \langle p_i, \hat{p}_{i,k} - p_i \rangle. \quad (75)$$

The velocity of a  $\beta$ -agent is related to  $p_i$  via the following equation

$$\hat{p}_{i,k} = \mu P p_i$$

where  $P = I - \mathbf{a}_k \mathbf{a}_k^T$  and  $\mathbf{a}_k$  is the unit normal to the surface of the obstacle. Moreover, for a wall  $\mu = 1$  and for a sphere  $\mu \in (0, 1]$ . In both cases, the following inner product is always non-positive

$$\langle p_i, \hat{p}_{i,k} - p_i \rangle = \mu (p_i^T P p_i) - p_i^T p_i = - \{ \mu (\mathbf{a}_k^T p_i)^2 + (1 - \mu) \|p_i\|^2 \} \leq 0.$$

Due to non-negativity of the heterogeneous adjacency elements  $b_{i,k}$ , we get  $\dot{K}_r \leq 0$ .  $\square$

## 7.6 Analysis of Flocking with Obstacle Avoidance

A natural question is whether the particle system obtained by applying Algorithm 3 is dissipative. The answer in this case is not as predictable as the case of interactions among  $\alpha$ -agents. The reason is that in free-flocking, every  $\alpha$ -agent reciprocates the action of its neighboring  $\alpha$ -agents, but in constrained flocking the  $(\alpha, \beta)$ -net is a unidirectional graph (or digraph).

**Theorem 6.** *Consider a particle system applying Algorithm 3 (or protocol (67)). Assume that the  $\gamma$ -agent is a static agent with a fixed state  $(q_r, p_r) = (q_d, p_d)$ . Define the energy function  $H(q, p) = V(q) + T(q, p)$  with kinetic energy  $T(q, p) = \frac{1}{2} \sum_{i=1}^n \|p_i\|^2$ . Suppose there exists a finite time  $t_0 \geq 0$  such that the average velocity of all agents satisfies the condition*

$$\frac{n}{2} \langle p_c(t), p_d \rangle \leq T(q(t), p(t)), \quad \forall t \geq t_0. \quad (76)$$

*Then, the energy of the system is monotonically decreasing (i.e.  $\dot{H}(q(t), p(t)) \leq 0$ ) along the trajectory of the collective dynamics of the multi-species system for all  $t \geq t_0$ .*

**Proof.** By direct differentiation, we have

$$\begin{aligned} \dot{H}(q, p) &= \langle \nabla V_\alpha(q), p \rangle + \langle \nabla_q V_\beta(q), p \rangle + \langle \nabla V_\gamma(q), p \rangle + \langle \nabla_{\hat{q}} V_\beta(q), \hat{p} \rangle + \sum_{i \in \mathcal{V}_\alpha} \langle p_i, u_i^\alpha + u_i^\beta + u_i^\gamma \rangle \\ &= c_1^\beta \langle \nabla_{\hat{q}} V_\beta(q), \hat{p} \rangle + c_2^\alpha \sum_{(i,j) \in \mathcal{E}_\alpha(q)} a_{ij}(q) \langle p_i, p_j - p_i \rangle \\ &+ c_2^\beta \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} b_{i,k} \langle p_i, \hat{p}_{i,k} - p_i \rangle - c_2^\gamma \sum_{i \in \mathcal{V}_\alpha} \langle p_i, p_i - p_d \rangle. \end{aligned}$$

But

$$\langle \nabla_{\hat{q}} V_\beta(q), \hat{p} \rangle = \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} \langle \nabla_{\hat{q}_{i,k}} V_\beta(q), \hat{p}_{i,k} \rangle = \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} \phi_\beta(\|q_j - q_i\|_\sigma) \langle \hat{n}_{i,k}, \hat{p}_{i,k} \rangle = 0, \quad (77)$$

because  $\hat{p}_{i,k}$  is tangent to the surface of the obstacle, whereas  $\hat{n}_{i,k}$  is orthogonal to the surface of the obstacle. Thus,  $\dot{H}(q, p)$  satisfies

$$\dot{H}(q, p) = -c_2^\alpha (p^T \hat{L}(q) p) + c_2^\beta \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^\beta} b_{i,k} \langle p_i, \hat{p}_{i,k} - p_i \rangle - 2c_2^\gamma (T(q, p) - \frac{n}{2} (p_d^T \cdot p_c)) \leq 0, \quad \forall t \geq t_0 \quad (78)$$

where the second term in the last inequality is negative semidefinite based on Lemma 4 and the term  $T(q, p) - (n/2)(p_d^T \cdot p_c)$  is by assumption non-negative for all  $t \geq t_0$ .  $\square$

The interpretation of condition (76) for a group of particles with equal velocities is interesting. In this case,  $p_c = p_i$  for all  $i$ , and therefore (76) reduces to the inequality  $p_c^T p_d \leq \|p_c\|^2$ . Let  $\theta_{c,d}$  denote the misalignment angle between vectors  $p_c$  and  $p_d$  in  $\mathbb{R}^m$ , i.e.  $\cos(\theta_{c,d}) = \langle p_c \cdot p_d \rangle / (\|p_c\| \cdot \|p_d\|)$ . Suppose  $p_c, p_d \neq 0$ , then the group has to be sufficiently agile, or  $\|p_c\| \geq v_0 := \|p_d\| \cos(\theta_{c,d})$ . Intuitively, this can be interpreted as a collective effort by the group to keep up with the desired velocity  $p_d$ . For a  $\gamma$ -agent with  $p_d = 0$ , condition (76) trivially holds.

Analysis of an equilibrium state of a group of dynamic agents that perform flocking in presence of obstacles makes less sense when the flock does not pass around all the obstacles. To be more precise, it is less interesting to analyze the stability of the equilibrium of collective dynamics of a flock while some  $\beta$ -agents are permanently present. This is certainly not the case for problems such as *sensor placement* and *distributed sensing* [5]. On the other hand, if one assumes that after some finite time  $t_1 > 0$ , no  $\alpha$ -agent ever comes near an obstacle, the case reduces to analysis of free-flocking that has already been presented. Hence, we postpone a comprehensive analysis of the behavior of flocks in permanent presence of obstacles for mobile sensor networks to a future occasion.

## 7.7 Information Flow in Constrained Flocking: A Peer-to-Peer Network

The information flow in flocking with obstacle avoidance has a natural hierarchical architecture as shown in Fig. 7 (a). A  $\gamma$ -agent has the role of a *virtual-leader* (or commander) in charge of navigation and control of the behavior of a flock as a whole. As a result, the hierarchy in Fig. 7 (a) can be referred to as a *virtual-leader/follower architecture*.

The dashed line between the  $\gamma$ -agent and all  $\alpha$ -agents indicates a single information exchange at  $t = 0$  (otherwise, the algorithm becomes centralized). Note that a virtual-leader/follower architecture should not be confused with a leader/follower architecture in which the leader is one of the physical agents (e.g. a vehicle in a multi-vehicle system or a fish in a school).

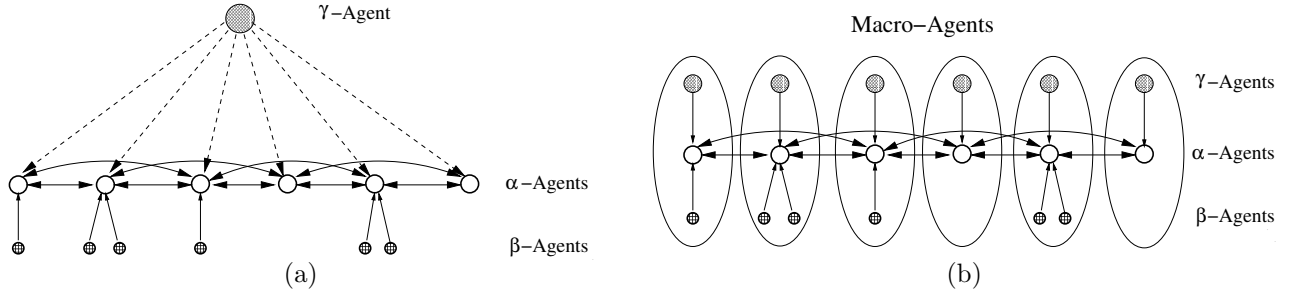


Figure 7: The in-agent and intra-agent information flow in constrained flocking: (a) a virtual-leader/follower hierarchical architecture and (b) a peer-to-peer architecture.

Since the computation required for implementation of virtual agents has to be carried out by embedded computers of a physical agent, Fig. 7 (a) *does not* provide a realistic picture of the computational architecture necessary for implementation of Algorithm 3. Though, the hierarchical architecture is useful in understanding why a  $\gamma$ -agent plays the role of a unifying objective that brings all the  $\alpha$ -agents together and assembles a connected network of mobile agents.

To model the information flow of Algorithm 3, we create one  $\gamma$ -agent corresponding to each  $\alpha$ -agent as shown in Fig. 7 (b). The new architecture is a *peer-to-peer network* that represents the interactions of a group of *macro-agents* (see Fig. 8). Each macro-agent consists of an  $\alpha$ -agent and its corresponding  $\gamma$ - and  $\beta$ -agents as illustrated in Fig. 8. The dynamics of the macro-agent of Algorithm 3 is highly structured (Fig. 8 (a)). Fig. 8 (c) demonstrates that Algorithms 1 & 2 are special cases of Algorithm 3 and can be implemented using a peer-to-peer network.

In this network of macro-agents, two macro-agents only communicate the state of their public components (i.e.  $\alpha$ -agents). Under the assumption that the initial state and dynamics of all  $\gamma$ -agents are equal, the virtual-leader/follower and peer-to-peer architectures become equivalent representations of a multi-species particle system.

The biological implication of feasibility of performing tracking/migration for groups of dynamic agents using a peer-to-peer network is that “*flocks need no leaders*”. This mathematically confirms a fact that has been known to animal behavior scientists for years [47].

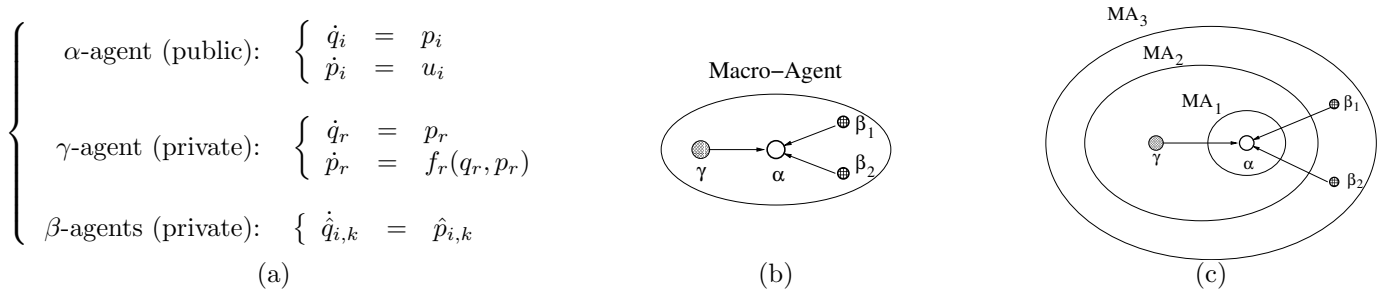


Figure 8: Macro-agent: (a) dynamics, (b) information flow among species for Algorithm 3, and (c) nested diagrams of macro-agents for Algorithms 1, 2, 3.

## 7.8 Conflict of Tasks in Constrained Flocking

It is not difficult to imagine that given a non-convex obstacle  $O_k$ , there might be multiple  $\beta$ -agents on  $O_k$  that are equally close to a generating  $\alpha$ -agent. Assuming that all of these  $\beta$ -agents repel this  $\alpha$ -agent, one might expect that in certain situations there would be a *conflict of tasks* between obstacle avoidance by a group of  $\alpha$ -agents and



achieving a group objective specified by a  $\gamma$ -agent. An example of this scenario is shown in Fig. 9 (a). In this case, a group of  $\alpha$ -agents get trapped and achieve only one of their two conflicting objectives.

The same scenario can occur in presence of a convex obstacle as well. This is illustrated in Fig. 9 (b). In this case, if the agents pass through the wall and reach their destination, they have clearly violated their obstacle avoidance objective. If they stay behind the wall and avoid collision to the wall, they can no longer reach their destination as specified by the  $\gamma$ -agent. This raises a *feasibility question* to make sure that no conflict of tasks exist for some agents during constrained flocking.

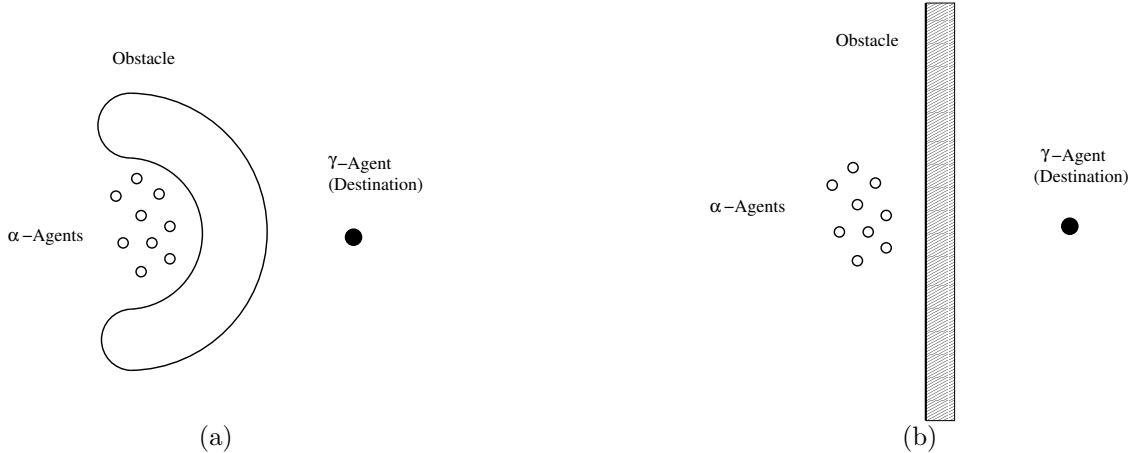


Figure 9: Trapped  $\alpha$ -agents due to conflicting objectives: (a) a non-convex obstacle and (b) a convex obstacle.

The use of a more sophisticated obstacle avoidance approach eliminates getting trapped behind an obstacle. A successful example of such an approach is the use of *gyroscopic forces* for obstacle avoidance in [3, 4]. For a comprehensive treatment of obstacle avoidance in robotics, please refer to Latombe [20].

## 8 Simulation Results

In this section, we present several simulation results of 2-D and 3-D flocking. For every case, a computer animation is available upon request. The following parameters remain fixed throughout all simulations:  $d = 7$ ,  $r = 1.2d$  (or  $\kappa = 1.2$ ),  $d' = 0.6d$ ,  $r' = 1.2d'$ ,  $\epsilon = 0.1$  (for  $\sigma$ -norm),  $a = b = 5$  for  $\phi(z)$ ,  $h = 0.2$  for the bump function of  $\phi_\alpha(z)$ ,  $h = 0.9$  for the bump function of  $\phi_\beta(z)$ , and the step-size in all simulations ranges between 0.01 to 0.03 seconds (equivalent to an update frequency of 33 Hz to 100 Hz). The parameters of the flocking algorithms and the types of the initial states are specified separately for each experiment. The set of  $l$  spherical obstacles are specified with an  $(m + 1) \times l$  matrix  $M_s$  where each column of  $M_s$  is the vector  $\text{col}(y_k, R_k) \in \mathbb{R}^{m+1}$ .

In all simulation results, the heading angle (or attitude) of each  $\alpha$ -agent specifies the direction of the velocity of that agent. In addition, the position of a dynamic  $\gamma$ -agent is marked with a  $\times$  sign.

### 8.1 2-D Flocking in Free-Space ( $n = 150$ )

Fig. 10 shows consecutive snapshots of 2-D flocking for 150 agents in free-space using Algorithm 2. The initial positions are chosen randomly from a Gaussian distribution with variance of 2500. The initial velocity coordinates are uniformly chosen at random from the box  $[-2, -1]^2$ . A flock is formed in Fig. 10 (d) and maintained thereafter. A dynamic  $\gamma$ -agent is used for this example. Fig. 11 demonstrates the framenets corresponding to the conformations of the agents given in Fig. 10. The number of edges of the dynamic graph  $G(q(t))$  increases by time and has a tendency to render the net connected. The set of initial positions are chosen uniformly at random so that the net  $G(q(0))$  is highly disconnected (i.e. has too many components). This makes the task of flocking more challenging. The planarity of the framenets in Figs. 10 (c) through (f) is very clear. Numerical measurements indicate that the final conformation is a low-energy quasi  $\alpha$ -lattice that induces a flock (see Fig. 18 (a)). These observations are in close agreement with our theoretical predictions in Section 5.

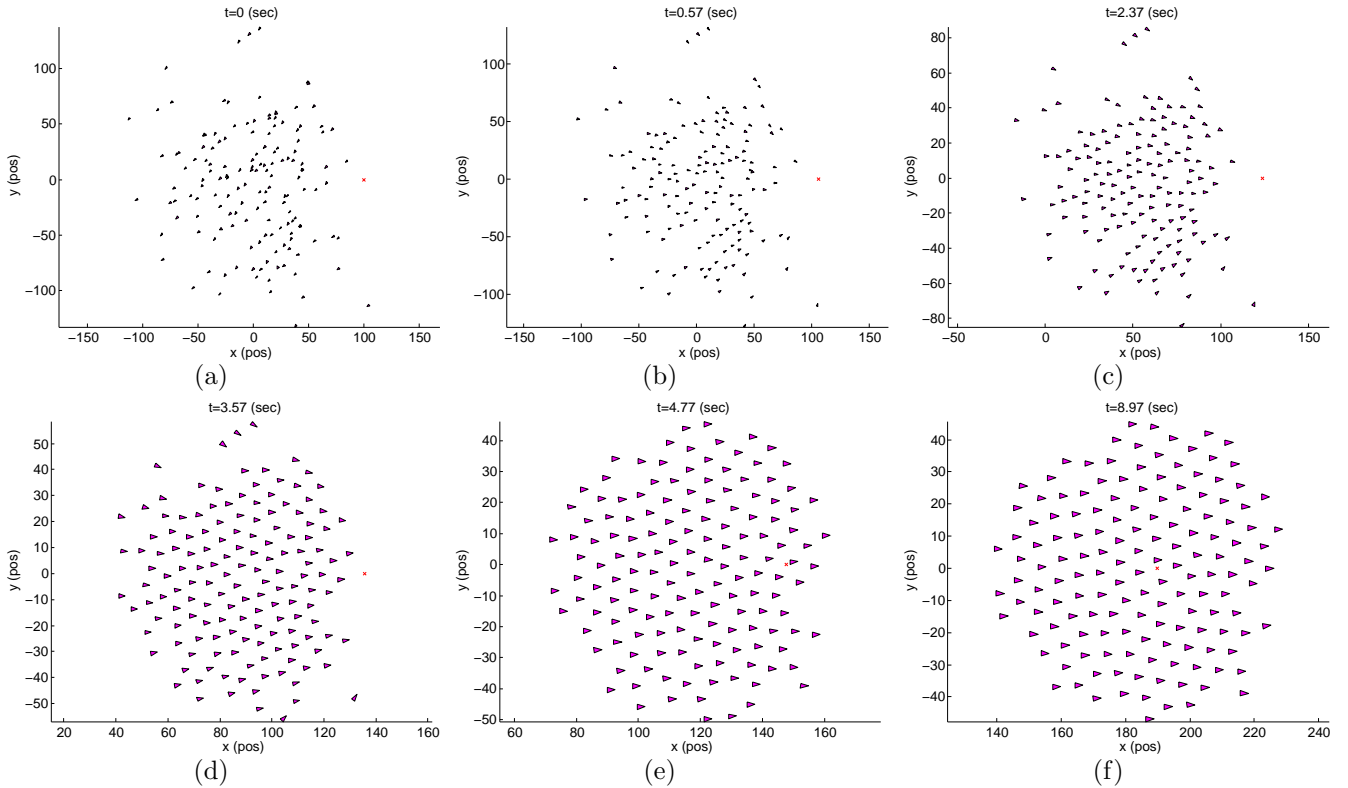


Figure 10: 2-D flocking for  $n = 150$  agents.

## 8.2 2-D Fragmentation in Free-Space ( $n = 40$ )

Fig. 12 demonstrates the fragmentation phenomenon for  $n = 40$  agents applying Algorithm 1. It is not surprising that with a random set of initial states, flocking behavior is not created. In Fig. 12 (f), one can identify 9 distinct small flocks (each contains at list two agents) and three individual agents.

In Algorithm 1 (or the  $(\alpha, \alpha)$  protocol), no  $\gamma$ -agent exists due to lack of existence of a group objective. This simulation result is another evidence that demonstrates creation of flocking motion is rather nontrivial because the idea of using a gradient-based control law plus a velocity matching term does not necessarily work!

Fragmentation is a generic form of collective behavior of agents applying Algorithm 1. This behavior is insensitive to the type of probability distribution of the initial position of the agents. Apparently, for the case of a highly dense initial net with small initial velocity mismatch, one might expect that the group of agents form a giant component (i.e. a connected subgraph with a large order).

In Figs. 12 (d) through (f), it can be observed that two agents which belong to two different flocks, move further apart from each other as time goes by. Again, this is a generic property of fragmentation. Fragmentation phenomenon can be viewed as lack of cohesion in a group of particles (see Fig. 18 (b)).

## 8.3 3-D Flocking in Free-Space: Automated Rendezvous

Fig. 13 shows the consecutive snapshots of 3-D flocking for a group of  $n = 50$  agents using Algorithm 2. Each agent represents a UAV moving in  $\mathbb{R}^3$ . The initial state of the agents is chosen at random with a Gaussian distributed. The attitude of each UAV is a rotation matrix  $R_i \in SO(3)$  with columns  $r_1, r_2, r_3$ . We use the following steps to determine  $R_i$  from velocity  $p_i \neq 0$ : i) set  $r_1 = p_i / \|p_i\|$  and let  $e_3 = (0, 0, 1)^T$ , ii) set  $r_2 = e_3 \times r_1$ , and iii) set  $r_3 = r_1 \times r_2$  ( $\times$  denotes the cross-product in  $\mathbb{R}^3$ ). For  $p_i = 0$ , define  $R_i = I_3$ . Based on this experiment, flocking can be used as a means of *automated rendezvous* (or gathering) for a medium to large number of autonomous agents. It is apparent that after some finite time, the agents self-assemble a connected network and maintain its connectivity thereafter. The formal proof of this statement requires the proof of Conjecture 1 in Section 5.

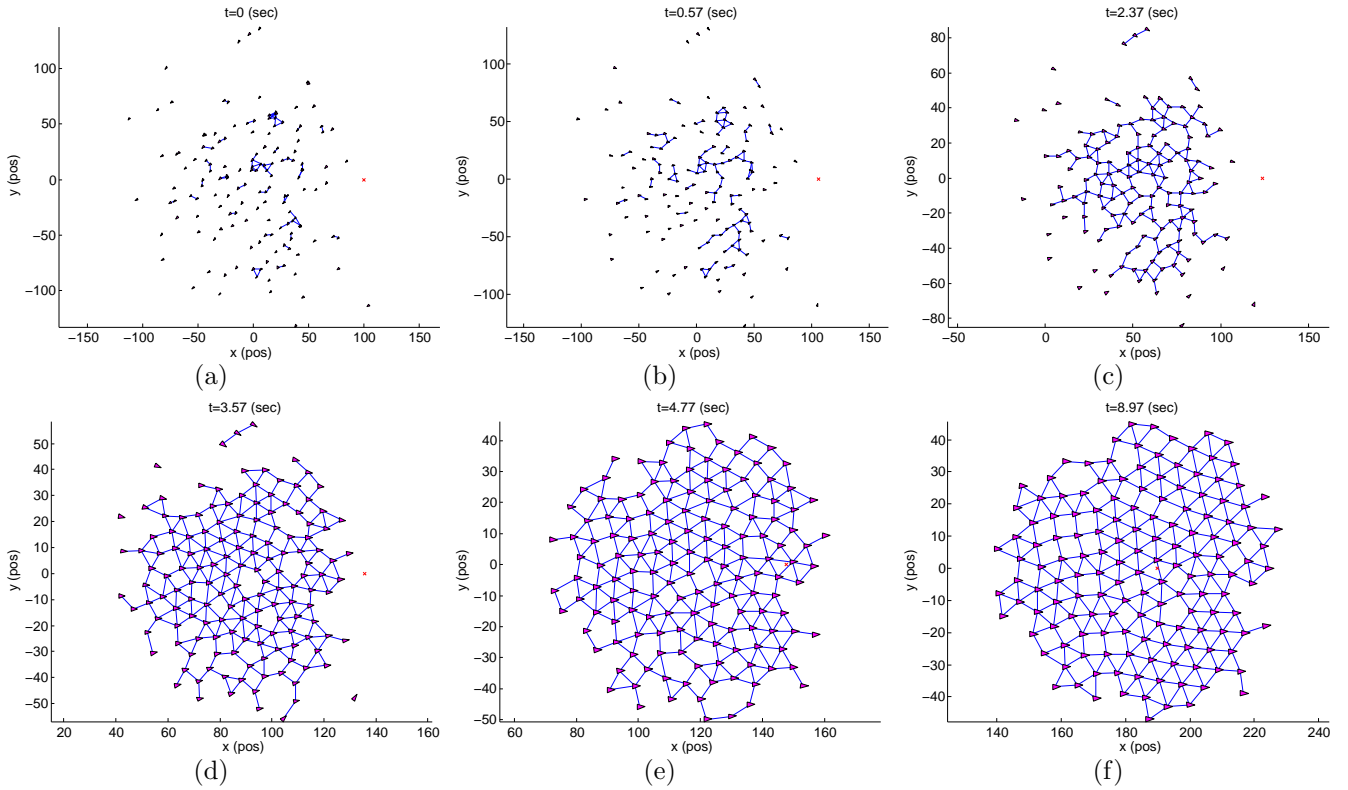


Figure 11: 2-D flocking for  $n = 100$  with a dynamic topology.

#### 8.4 Split/Rejoin Maneuver: Low-Altitude Flight of UAVs and Predator Evasion

Consider a group of agents that intend to move/migrate from point  $A$  to  $B$ . Here,  $A$  and  $B$  are the positions of the center of mass (CM) of the group at the group's source and destination. Whenever there are multiple obstacles along the straight line connecting  $A$  to  $B$ , the agents cannot pass through the obstacles. As a result, they might split in two or multiple smaller groups. This maneuver is useful for missions that require low-altitude flight of UAVs or moving in urban environments. In schools of fish, the split/rejoin maneuver is used as a predator evasion tactic [36].

The objective in performing a *split/rejoin maneuver* is to gather various groups that have initially split due to the presence of obstacles or adversarial agents. The split/rejoin maneuver is demonstrated in Figs. 14 and 15 for a group of  $n = 150$  agents in presence of  $l = 6$  obstacles. Fig. 15 demonstrates the evolution of the topology of inter-agent interactions. Based on this figure, it is clear that the topology of a network of dynamic agents during flocking undergoes frequent changes. In other words, flocking involves stability analysis for a network of dynamic agents with switching topology (as in [33]).

The initial position of the agents are chosen uniformly at random from the box  $[-40, 80]^2$ . The initial velocities are set to zero. The group objective is specified by a static  $\gamma$ -agent with  $q_d = (200, 30)^T$  and  $p_d = (5, 0)^T$ . Moreover,  $c_1^\alpha < c_1^\gamma < c_1^\beta$  and  $c_2^\gamma = 2\sqrt{c_1^\gamma}$  for all species. The matrix of obstacles is

$$M_s = \begin{bmatrix} 100 & 110 & 120 & 130 & 150 & 160 \\ 20 & 60 & 40 & -20 & 40 & 0 \\ 10 & 4 & 2 & 5 & 5 & 3 \end{bmatrix}.$$

Based on Fig. 15, after all agents pass the obstacles on their way, the group forms a giant flock. It was numerically verified that no agent ever entered any of the six obstacles for the complete trajectory of the particles. Fig. 14 illustrates the split/rejoin maneuver without drawing the edges of the  $(\alpha, \beta)$ -net at each moment.

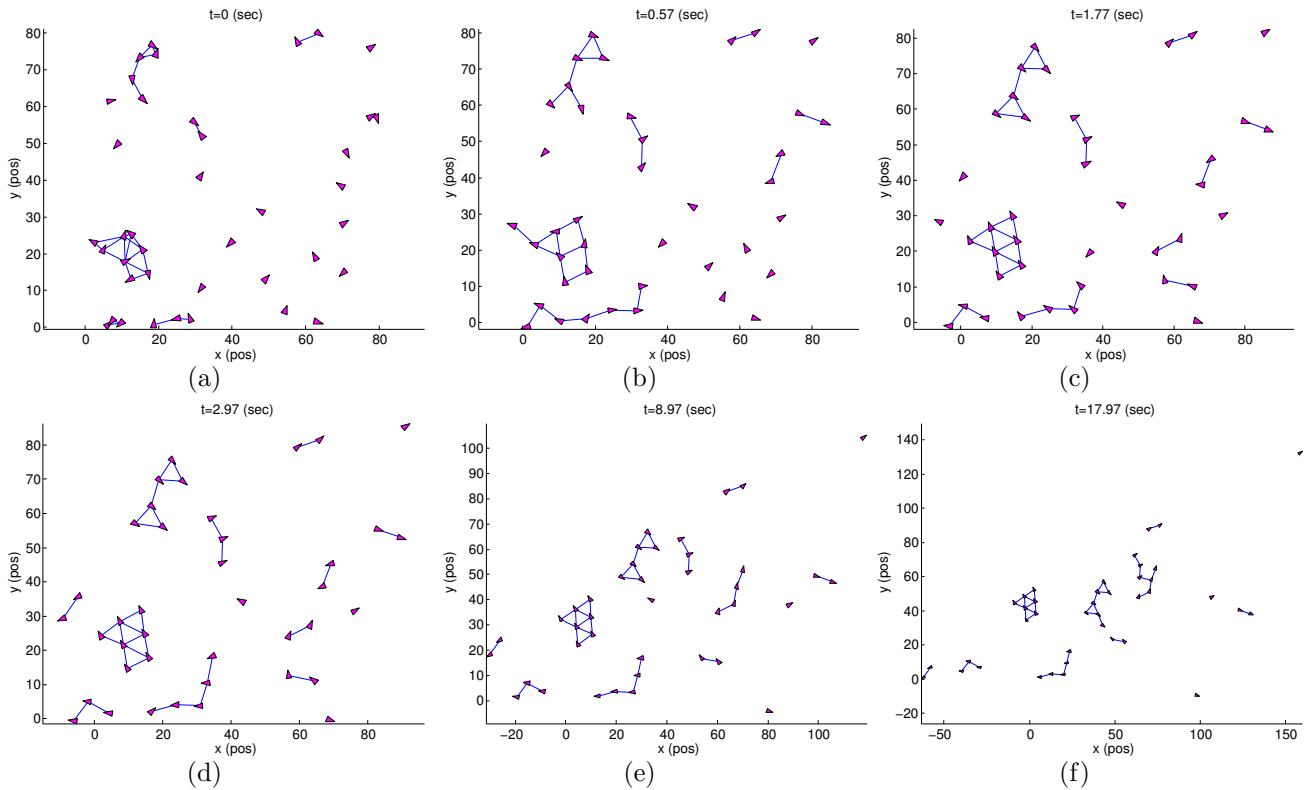


Figure 12: The fragmentation phenomenon for 40 agents applying Algorithm 1.

## 8.5 Squeezing Maneuver: Moving through Narrow Spaces

The *squeezing maneuver* is the result of flocking in presence of obstacles that are relatively close to each other. This is a special case of escape panic phenomenon. By “relatively close”, we mean that the narrow pathway between the obstacles is about  $2r$  to  $3r$  wide. Figs. 16 and 17 illustrate the task of squeezing maneuver for  $n = 150$  agents. The initial positions of the agents are  $n = 150$  random points that are uniformly distributed in the box  $[0, 120]^2$ . The initial velocity of all agents is set to zero. For this case, the group objective is specified by a static  $\gamma$ -agent with  $q_d = (230, 60)^T$  and  $p_d = (6, 0)^T$ . In addition,  $c_1^\alpha < c_1^\beta < c_1^\gamma$  and  $c_2^\nu = 2\sqrt{c_1^\nu}$  for  $\nu = \alpha, \beta, \gamma$ . The matrix of obstacles is given by

$$M_s = \begin{bmatrix} 150 & 150 \\ 30 & 100 \\ 25 & 25 \end{bmatrix}$$

According to Fig. 17, one can observe that the agents avoid collision with both obstacles as moving forward. This has been numerically verified for the entire trajectory of the particles. Since the desired group velocity is non-zero, the group does not stop near  $q_d$  and moves along the specified desired group velocity  $p_d$ . After passing both obstacles, the agents form a giant flock as shown in Fig. 17 (f).

## 9 What Constitutes Flocking?

In [35], Partridge provides a brief survey of various definitions of “schooling in fish” by animal behavior scientists that spans half a century from 1927 to 1981. The length of this period is a clear indication of the difficulty of the task in hand. To give an objective definition of flocking, we determine a quantitative measure of flocking that is independent of collective dynamics of the agents. In the sense that it does not depend on a specific method used for generation of trajectories of particles, i.e. the measure is *universal*. In the following, we define a special form of flocking called  $\alpha$ -flocking:

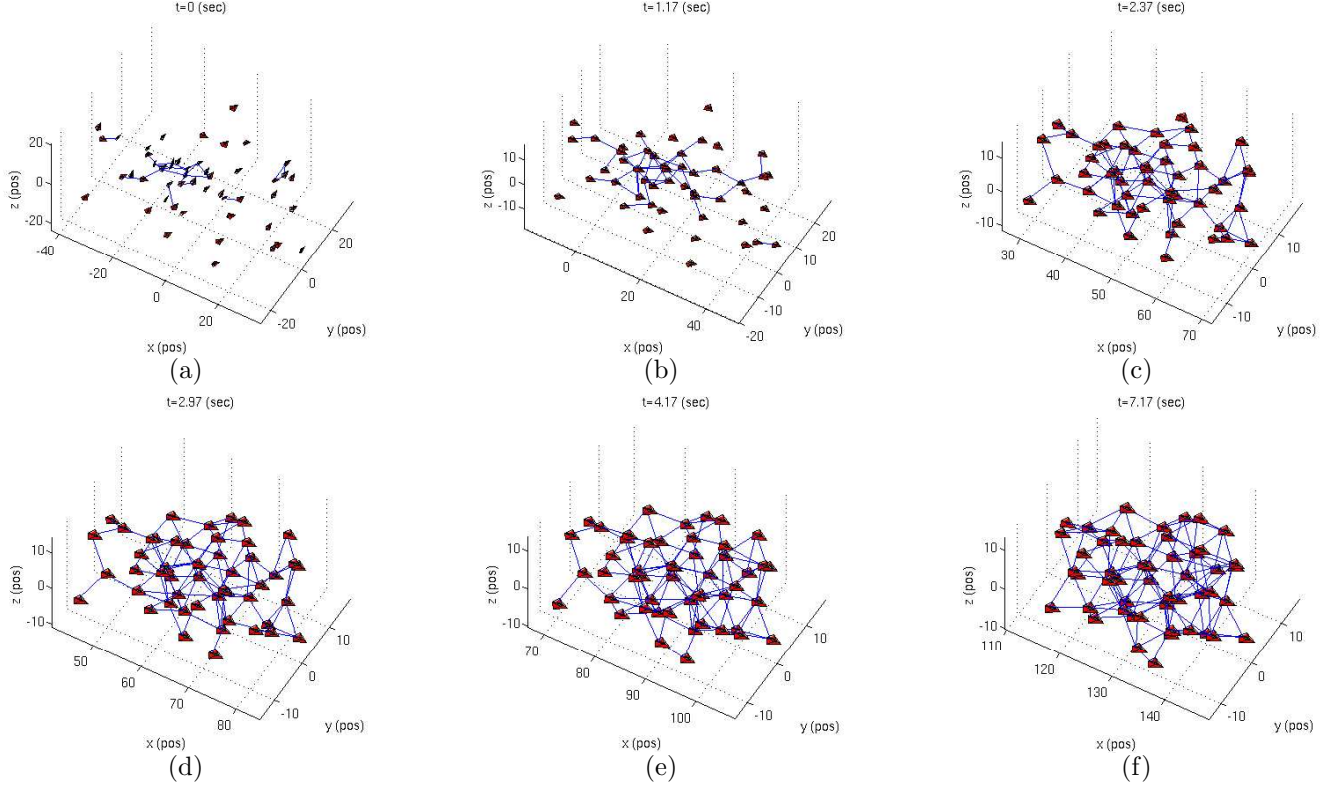


Figure 13: Snapshots of 3-D flocking/automated rendezvous using Algorithm 2 for  $n = 50$  UAVs.

**Definition 5.** ( $\alpha$ -flocking) Let  $z : t \mapsto \text{col}(q(t), p(t))$  be the state trajectory of a system of  $n$  dynamic agents (or particles). We say a group of agents perform  $\alpha$ -flocking over the time interval  $[t_0, t_f]$  if there exists relatively small numbers  $\epsilon_0, \epsilon_1, \epsilon_2 > 0$  and a distance  $d > 0$  such that the trajectory  $z(t)$  satisfies all the following conditions for all  $t \in [t_0, t_f]$  with an interaction range  $r = (1 + \epsilon_0)d$ :

- i) The group remains a quasi-flock (i.e.  $G(q(t))$  has a giant component).
- ii) The group remains cohesive.
- iii) The deviation energy remains small (or  $E(q(t)) \leq \epsilon_1 d^2$ ).
- iv) The velocity mismatch remains small (or  $K(v(t)) \leq \epsilon_2 n$ ).

A more strict form of flocking, or *strict  $\alpha$ -flocking*, can be defined by replacing the above four conditions with the following three properties:

- a) The group remains a flock (i.e. net  $G(q(t))$  is connected).
- b) The deviation energy remains small (i.e.  $E(q(t)) \leq \epsilon_1 d^2$ ).
- c) The velocity mismatch remains small (i.e.  $K(v(t)) \leq \epsilon_2 n$ ).

One can use conditions a) and b) to establish that the group remains cohesive over the interval  $[t_0, t_f]$ . Definition 5 has the same role for flocking in particle systems that Lyapunov stability has for nonlinear dynamical systems. It is worthwhile mentioning that “regular fragmentation” violates conditions i), ii), and iv) of Definition 5, “irregular fragmentation” violates all four conditions of  $\alpha$ -flocking, and “irregular collapse” severely violates condition iii). Therefore, irregular/regular fragmentation and irregular collapse do not constitute  $\alpha$ -flocking as mentioned in Section 1. Furthermore, “regular collapse” is an acceptable form of  $\alpha$ -flocking for when a comfortable/safe inter-agent

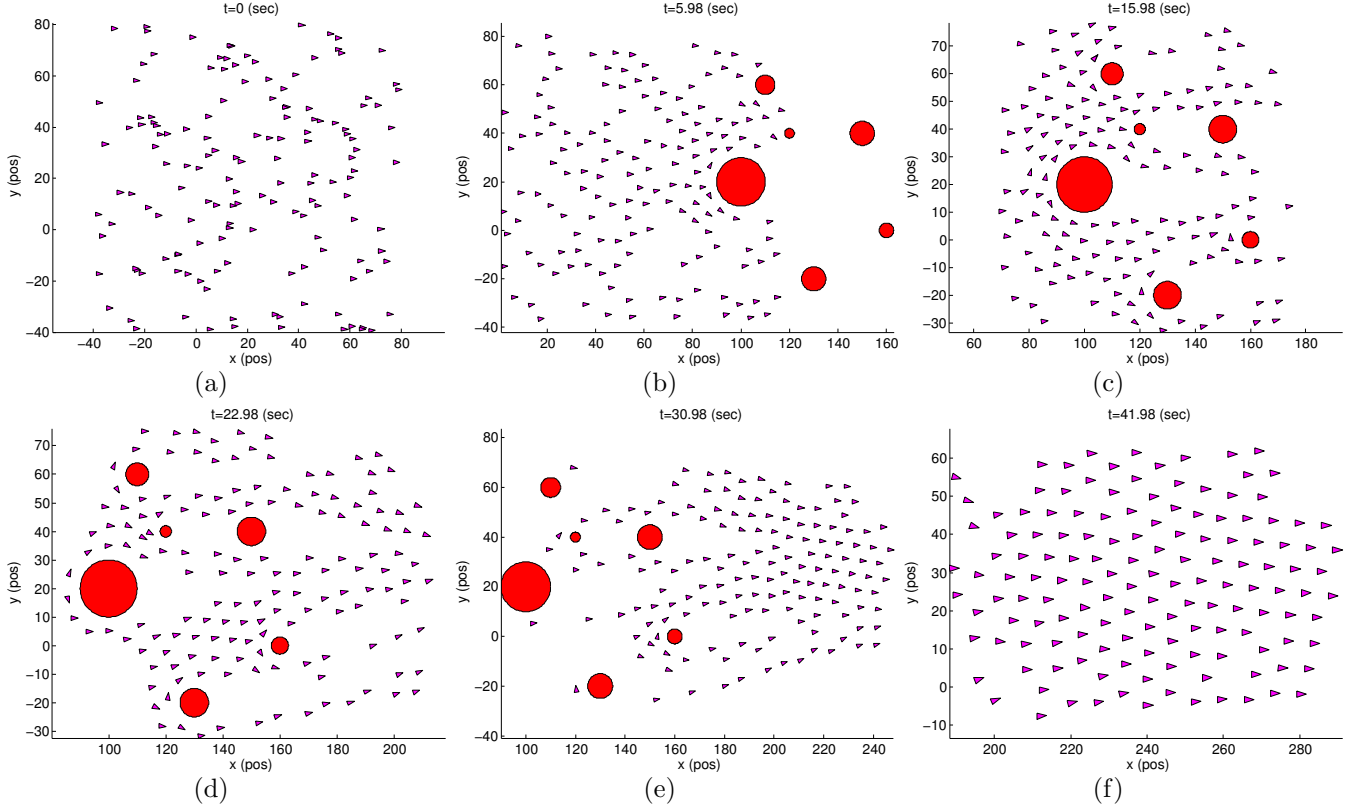


Figure 14: The split/rejoin maneuver for  $n = 150$  agents.

distance  $d$  is replaced by an uncomfortable distance  $d' \ll d$ . Surprisingly, regular collapse phenomenon can be found in nature as a defense mechanism used by schools of fish.

*Formation flight* (e.g. for birds) can be viewed as the *most strict form of  $\alpha$ -flocking* with  $\epsilon_1 = \epsilon_2 = 0$  and a fixed topology  $G(q(t)) = G(t_0), \forall t \in [t_0, t_f]$ . This is consistent with the prediction of Theorem 3 (part ii) that asymptotically the topology of the network of agents evolves to a fixed graph  $G^* = G(x_\lambda^*)$ .

The main feature of Definition 5 is that  $\alpha$ -flocking can be numerically verified for the trajectories of a system of particles regardless of the method of trajectory generation. Meaning that the definition of  $\alpha$ -flocking is *universal* (or algorithm-independent). This is analogous to universality of the definition of “Lyapunov stability” for nonlinear systems (Lyapunov stability is a property of the solutions of a nonlinear system and not its vector field). The challenge is to search for parameters  $\epsilon_0, \epsilon_1, \epsilon_2, d$ . This verification process is discussed in Section 9.1.

## 9.1 Verification of $\alpha$ -Flocking

To verify whether a group of particles perform  $\alpha$ -flocking, we need to calculate four quantities along the trajectory of the particles. These quantities are defined in the following:

- i) **Relative Connectivity:** Since the rank of Laplacian of a connected graph of order  $n$  is at most  $(n - 1)$ , we define the *relative connectivity* of the group at time  $t$  as  $C(t) = \frac{1}{n-1} \text{rank}(L(q(t))) \in [0, 1]$ , where  $L(q)$  is the adjacency matrix of a graph with 0-1 adjacency elements corresponding to the set of edges of the net  $G(q)$ .
- ii) **Cohesion Radius:** We define the *cohesion radius* of a group of agents at time  $t$  as  $R(t) = \max_{i \in \mathcal{V}} \|q_i(t) - q_c(t)\|$ . A cohesive group has a finite cohesion radius.
- iii) **Normalized Deviation Energy:**  $\tilde{E}(q) = E(q)/d^2$ .
- iv) **Normalized Velocity Mismatch:**  $\tilde{K}(v) = K(v)/n$ .

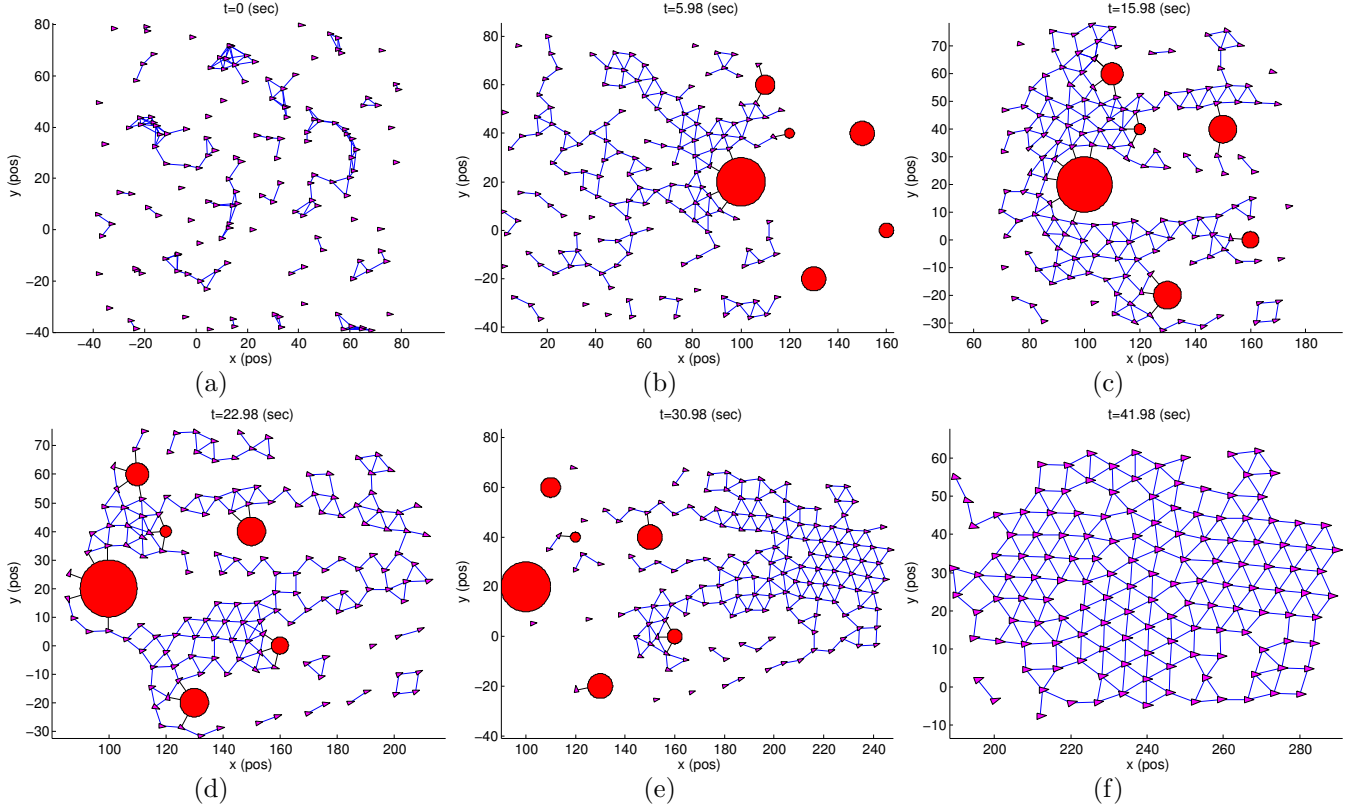


Figure 15: The split/rejoin maneuver for  $n = 150$  agents.

To clarify the use of these quantities, consider the trajectory of  $n = 150$  particles in simulation of 2-D flocking in Section 8.1 (Figs. 10, 11). The curves  $C, R, \bar{E}, \bar{K}$  for this trajectory are plotted in Fig. 18 (a) over the first 300 iterations. (The step-size is  $T = 0.03$  (sec) and  $t = kT$  where  $k$  is the iteration number.) At iteration  $k_c = 134$ , a phase transition occurs and the net  $G(q(t))$  becomes connected. Clearly, the network topology remains connected for all future iterations  $k > k_c$ . The point  $k = k_c$  is shown by a dashed line. The effect of this phase transition is clear in the (normalized) velocity mismatch curve after  $k = 170$ . In this case, the cohesion radius is monotonically decreasing. It is also clear that the (normalized) energy deviation remains relatively small after  $k = 170$  iterations. Based on the simulation data, we conclude that  $\alpha$ -flocking is achieved after the 170<sup>th</sup> iteration with parameters  $\epsilon_0 = 0.2$ ,  $\epsilon_1 = 0.005$ ,  $\epsilon_2 = 0.3$ .

In contrast, Fig. 18 (b) shows the four curves obtained from the trajectory of particles in Fig. 12. Here are some observations: 1) the cohesion radius is monotonically increasing, 2) the net never becomes connected and has too many components, 3) the velocity mismatch never reduces below a large constant, and 4) the deviation energy asymptotically vanishes implying conformation of  $\alpha$ -lattices. All these facts are consistent with the occurrence of the regular fragmentation phenomenon because only condition iii) of  $\alpha$ -flocking is satisfied.

A better measure of relative connectivity is

$$C^* = \frac{|\mathcal{E}(\Gamma^*)|}{|\mathcal{E}(G)|} \quad (79)$$

where  $\Gamma^*$  is the connected subgraph of  $G$  with the highest number of nodes. We refer to  $C^*$  as the *semi-connectivity factor* of a graph. The advantage of  $C^*$  over  $C$  is the ability to distinguish fragmentation phenomenon from flocking with high confidence. For example, in a quasi-flock of  $n = 100$  agents that consists of 5 components with population of 1, 1, 1, 1, 96, we have  $C_1^* = 0.96$  and  $C_1 = 95/99$ , whereas in a group of 5 flocks each with 20 agents  $C_2^* = 0.2$  and  $C_2 = 95/99$ . The separation gap of the semi-connectivity factor is  $\Delta C^* = |C_1^* - C_2^*| = 0.76$ . This is a very large gap compared to the zero gap obtained from the ranked-based factor  $C$ .

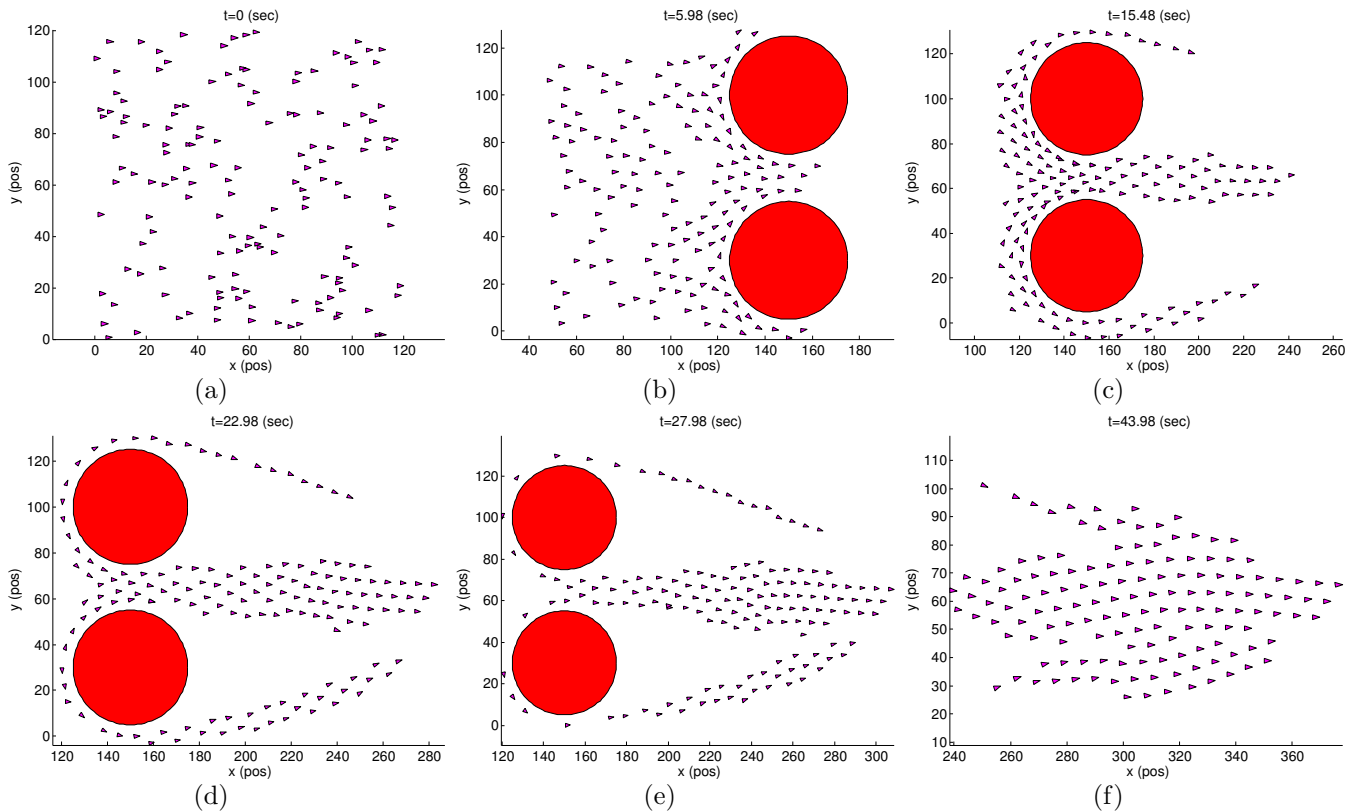


Figure 16: The squeezing maneuver for  $n = 150$  agents.

A second example is the fragmentation phenomenon shown in Fig. 12 where the final conformation has a semi-connectivity factor of  $C^* = 7/40 = 0.175 \ll 1$  and a ranked-base factor of  $C = 28/40 = 0.7$ . Clearly,  $C$  is deceptively high.

*Remark 9.* An algorithm is given in [4] that leads to creation of flocking behavior in simulations (no analysis of flocking is available for this algorithm). The foundation of this algorithm has to do nothing with all three algorithms presented in this paper. I would like to conjecture that the agents applying the flocking algorithm in [4] are capable of asymptotically performing  $\alpha$ -flocking (i.e.  $\exists T > 0 : \alpha$ -flocking holds for the trajectory of the particles for all  $t > T$ ).

## 10 Conclusions

This paper provides a theoretical framework for design and analysis of distributed flocking algorithms for multi-agent dynamic systems. The cases of flocking in free-space and presence of obstacles were both considered. It turns out that  $\alpha$ -lattices and quasi  $\alpha$ -lattices are point sets with important geometric and graph theoretic properties that have a crucial role in construction of collective potential functions for flocking. These collective potentials were used for both derivation and analysis of all three flocking algorithms.

First, we discussed that the use of the  $(\alpha, \alpha)$  protocol alone generically leads to regular fragmentation. Then, Algorithm 2 was constructed from the  $(\alpha, \alpha)$  flocking protocol by adding a navigational feedback. This navigational feedback takes the group objective into account using a static/dynamic  $\gamma$ -agent. It was demonstrated that Algorithm 2 generically leads to flocking. Two conjectures were stated that provide further details regarding geometric and graph theoretic properties of the augmented potential function. This led to explanation of the existence of a spatial order in flocks and self-assembly of connected networks of agents. Using stress elements of nets, we demonstrated that the  $(\alpha, \alpha)$  flocking protocol (or Algorithm 1) encompasses all three rules of Reynolds.

For flocking with obstacle avoidance, each obstacle in the neighborhood of an  $\alpha$ -agent was represented using a  $\beta$ -agent. A collective potential was constructed by taking into account the interactions of all three species of



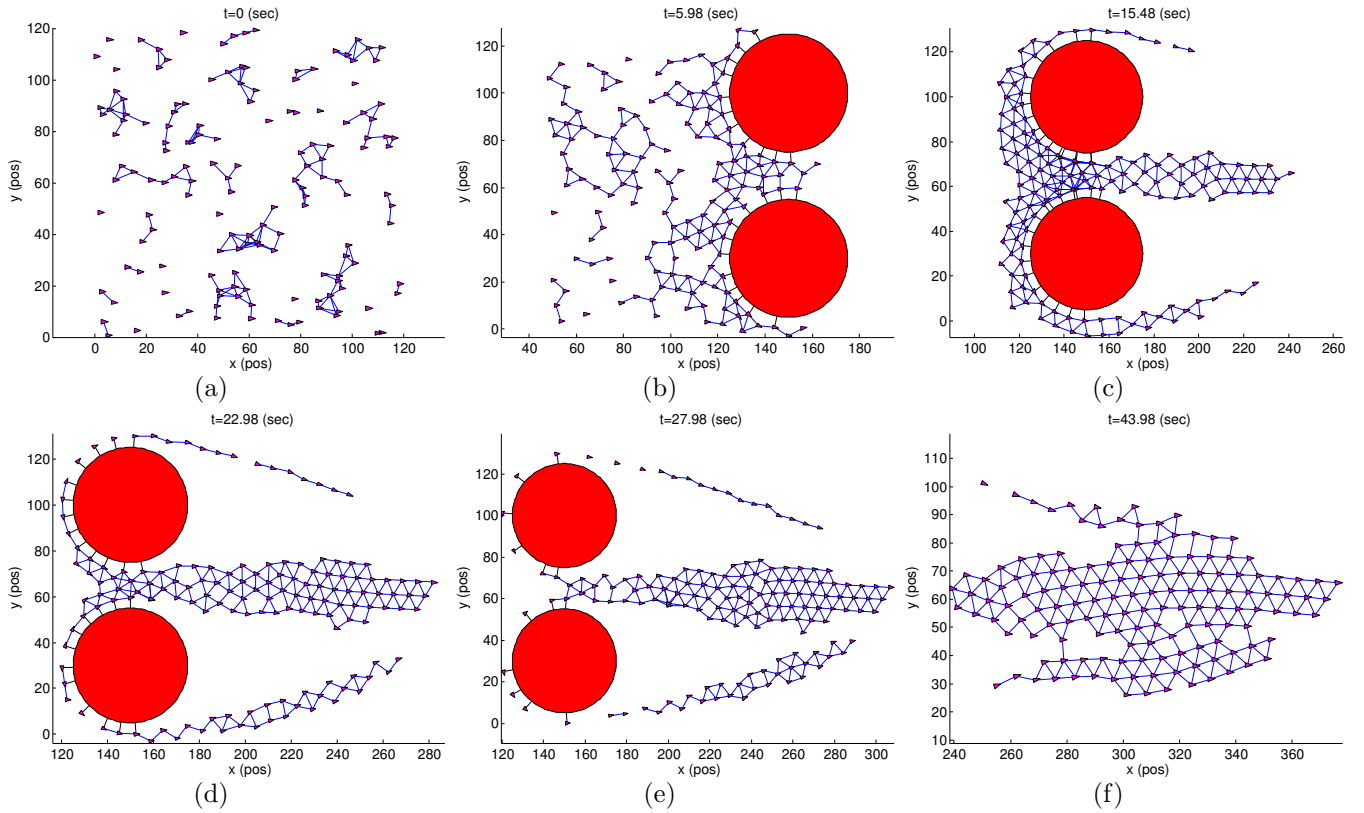


Figure 17: The squeezing maneuver for  $n = 150$  agents.

agents. Algorithm 3 was derived from this multi-species collective potential function. Both split/rejoin maneuver and squeezing maneuver were successfully performed using Algorithm 3.

We established that the architecture necessary for implementation of Algorithm 3 is a peer-to-peer network of macro-agents. The biological implication of this peer-to-peer architecture is that “flocks need no leaders”. A “universal” definition of flocking, called  $\alpha$ -flocking, was provided and a procedure for verification of  $\alpha$ -flocking was introduced. This definition is objective and does not depend on a particular method used for generation of the trajectories of particles.

## Acknowledgments

This research is supported in part by AFOSR under the grant F49620-01-1-0361 and by DARPA under the grant F33615-98-C-3613. Many thanks goes to Richard Murray and Jeff Shamma for their valuable comments.

## Appendix

### A Proofs of Some Results

#### A.1 Proof of Lemma 2

By definition,  $\dot{q}_c = p_c$  and  $\dot{x} = v$ . Given protocol (26), we have

$$\ddot{q} = -\nabla V(q) - \hat{L}(q)p + f_\gamma(q, p, q_r, p_r).$$

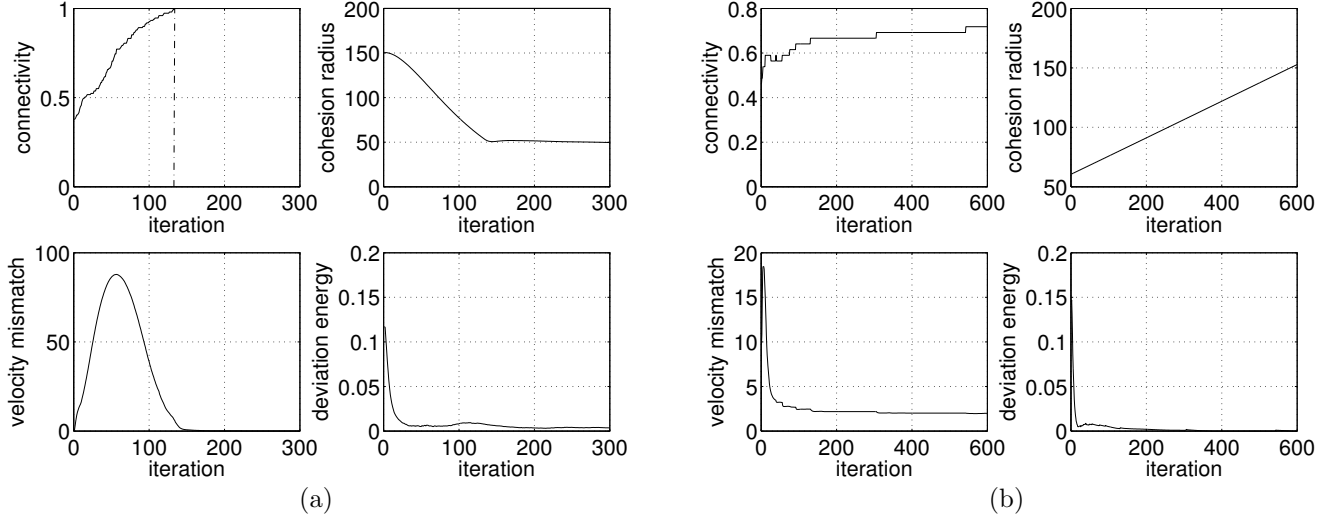


Figure 18: The  $C, R, \tilde{E}, \tilde{K}$  curves for simulations in Fig. 11 and Fig. 12: (a) flocking and (b) regular fragmentation.

Due to the fact that  $G(q)$  is an undirected graph,  $\text{Ave}(\nabla V(q)) = 0$  for all  $q$ . Similarly, we get  $\text{Ave}(f_d(q)) = \text{Ave}(\hat{L}(q)p) = 0$  for all  $q, p$ . Assume  $f_\gamma(q, p)$  is a linear controller in the form

$$f_\gamma(q, p, q_r, p_r) = -c_1(q - \mathbf{1}_n \otimes q_r) - c_2(p - \mathbf{1}_n \otimes p_r)$$

with  $\mathbf{1}_n = \text{col}(1, \dots, 1) \in \mathbb{R}^n$ . Then,  $f_\gamma(q, p)$  can be decomposed into two terms:

$$\begin{aligned} f_\gamma(q, p, q_r, p_r) &= -c_1(x + \mathbf{1}_n \otimes q_c - \mathbf{1}_n \otimes q_r) - c_2(v + \mathbf{1}_n \otimes p_c - \mathbf{1}_n \otimes p_r) \\ &= -c_1x - c_2v - c_1(\mathbf{1}_n \otimes (q_c - q_r)) - c_2(\mathbf{1}_n \otimes (p_c - p_r)) \\ &= g(x, v) + \mathbf{1}_n \otimes h(q_c, p_c, q_r, p_r) \end{aligned}$$

where  $g(x, v)$  and  $h(q_c, p_c, q_r, p_r)$  are defined in (35). Note that  $\text{Ave}(x) = \text{Ave}(v) = 0$  and thus  $\text{Ave}(g(x, v)) = 0$ . The translational dynamics can be obtained as follows

$$\ddot{q}_c = \text{Ave}(f_\gamma(q, p, q_r, p_r)) = \text{Ave}(g(x, v)) + \text{Ave}(\mathbf{1}_n \otimes h(q_c, p_c, q_r, p_r)) = h(q_c, p_c, q_r, p_r).$$

On the other hand,  $x = q - \mathbf{1}_n \otimes q_c$  and this gives

$$\begin{aligned} \ddot{x} &= -\nabla V(q) - \hat{L}(q)p + f_\gamma(q, p, q_r, p_r) - \mathbf{1}_n \otimes h(q_c, p_c, q_r, p_r) \\ &= -\nabla V(q) - \hat{L}(q)p + g(x, v). \end{aligned}$$

But  $x_j - x_i = q_j - q_i$  for all edges  $(i, j) \in \mathcal{E}(q)$  which means  $\nabla V(q) = \nabla V(x)$  and  $a_{ij}(q) = a_{ij}(x)$  for all edges  $(i, j)$ . Thus,  $\hat{L}(q) = \hat{L}(x)$  for all  $q$ . The first two terms of the expression for  $-\ddot{x}$  can be rewritten as

$$\begin{aligned} \nabla V(q) + \hat{L}(q)p &= \nabla V(x) + \hat{L}(x)(v + \mathbf{1}_n \otimes p_c) \\ &= \nabla V(x) + \hat{L}(x)v \end{aligned}$$

This is because any  $m$ -dimensional graph Laplacian  $\hat{W} = W \otimes I_m$  satisfies  $\hat{W}(\mathbf{1}_n \otimes y) = 0$  for all  $m$ -vectors  $y$ .

## A.2 Proof of Theorem 5

We prove parts i), ii), and iii) as three separate claims under the following assumption:

**Assumption 1.** Assume  $q$  is an  $\alpha$ -lattice with scale  $d > 0$  and ratio  $\kappa > 1$  that is collision-free in dimension  $m$  (i.e. no two nodes coincide).

**Claim 1.** Let  $q$  be an  $\alpha$ -lattice satisfying Assumption 1 for  $m = 2, 3$ . Then, the framenet induced by this  $\alpha$ -lattice is a planar graph.

**Proof.** The proof follows from elementary geometric arguments, yet it is nontrivial. Let  $(A, B)$  and  $(C, D)$  be two edges of a framenet  $(G(q), q)$  induced by an  $\alpha$ -lattice  $q$  of scale  $d$  and ratio  $\kappa = r/d > 1$ . Suppose that the partial lines  $AB$  and  $CD$  that satisfy  $AB = CD = d$  intersect at point  $P$  which is distinct from points  $A, B, C, D$  (See Fig. 19(a)). In three steps, we demonstrate that one reaches a contradiction.

Step 1: Assume  $(A, C), (B, C), (A, D), (B, D) \notin \mathcal{E}(q)$ . Without loss of generality, let

$$PB = \min\{PB, PA\}, \quad PC = \min\{PC, PD\},$$

Thus,  $AB \geq 2PB$  and  $CD \geq 2PC$ . We have

$$PB \leq d/2, \quad PC \leq d/2 \implies BC \leq PB + PC = d < r$$

Hence,  $B$  and  $C$  are neighbors and  $BC = d$  (because  $q$  is an  $\alpha$ -lattice). This only contradicts  $(B, C) \notin \mathcal{E}(q)$  and the rest of the assumptions still hold.

Step 2: From step 1, we have  $AB = BC = CD = d$  (see Fig. 19(b)). Assume  $A, C$  and  $B, D$  are not neighbors.

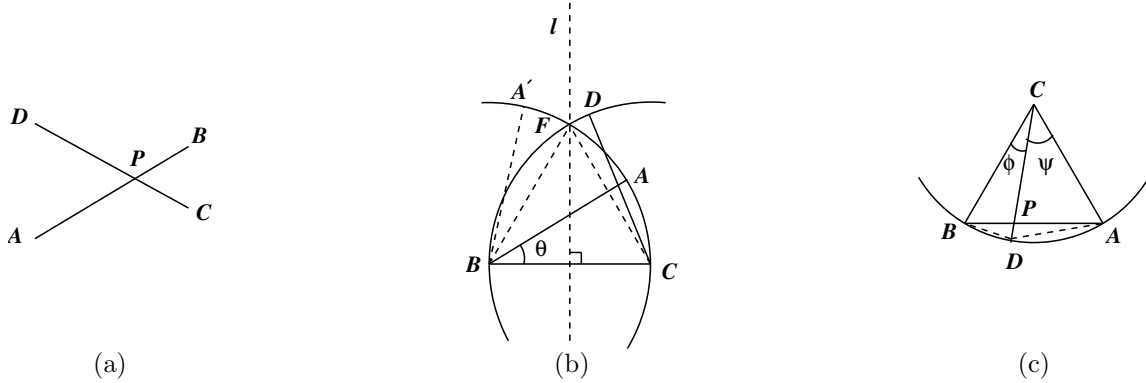


Figure 19: (a) two partial lines of equal length intersecting at  $P$ , (b) the edges  $(A, B), (B, C), (C, D)$  of equal length, and (c) an equilateral triangle  $ABC$  and a node  $D$  on the arc  $AB$ .

Draw two circles of radii  $d$  centered at points  $B$  and  $C$ . These circles intersect each other at  $F$  and  $\triangle FBC$  is an equilateral triangle. Points  $A$  and  $D$  are located on the circles centered at  $B$  and  $C$ , respectively, as shown in Fig. 19(b). Suppose  $A$  is not located on the arc  $FC$  and  $D$  is not located on the arc  $FB$ . Then,  $AB$  and  $CD$  never intersect because the line  $l$ , passing through  $F$  and orthogonal to  $BC$ , separates them. Unless,  $AB$  and  $CD$  meet at  $P = A = D$  which contradicts the assumption that partial lines  $AB$  and  $CD$  intersect at a point other than their endpoints. Hence, we assume  $A$  is located on the arc  $FC$  (without loss of generality). Let  $\theta = \angle ABC$ , then  $\theta \leq \pi/3$ . We have

$$AC = 2d \sin(\theta/2) \leq d < r,$$

which means  $A, C$  are neighbors and  $AC = d$ . Therefore,  $\triangle ABC$  is an equilateral triangle and  $A$  and  $F$  coincide (otherwise, if  $A$  and mirror image of  $F$  w.r.t.  $BC$  coincide, then  $AB$  and  $CD$  never intersect). Moreover, if  $D$  is anywhere but on the arc  $BF = BA$ ,  $AB$  and  $CD$  do not intersect. Hence,  $\triangle ABC$  is an equilateral triangle and  $D$  is on the arc  $BA$  (see Fig. 19(c)).

Step 3: Draw a circle of radius  $d$  centered at  $C$ . Clearly, this circle passes through  $A$  and  $B$ . Since  $CD$  intersects  $AB$  at  $P$ , the continuation of the partial line  $CP$  intersects the arc  $AB$  at point  $D$ . Set  $\phi = \angle DCB$  and  $\psi = \angle DCA$  and notice that  $0 < \phi, \psi < \pi/3$ . We get

$$AD = 2d \sin(\psi/2) < d < r.$$

Similarly, one obtains  $BD < r$ . Hence,  $\triangle DBA$  is an equilateral triangle with a side length of  $d$ . In other words, all four node  $A, B, C, D$  are mutually neighbors of each other and  $CD = \sqrt{3}d$ . This contradicts the assumption that  $CD = d$  and the result follows for dimension  $m = 2$ . The proof for dimension  $m = 3$  reduces to the preceding proof, once one passes a 2-D plane  $\pi$  through the intersection point  $P$  with a surface normal of  $\overrightarrow{AB} \times \overrightarrow{CD}$ .  $\square$

**Claim 2.** *Suppose Assumption 1 holds. Then, the net  $G(q)$  induced by an  $\alpha$ -lattice at most has  $3n - 6$  links (or undirected edges).*

**Proof.** This is a well-known fact in graph theory based on Euler's formula that any planar graph at most has  $3n - 6$  undirected edges [2, 7].  $\square$

**Claim 3.** *Suppose Assumption 1 holds. Then, the net  $G(q)$  with  $n > m + 1$  nodes is not a complete graph in dimensions  $m = 1, 2, 3$ .*

**Proof.** The proof is trivial for dimension 1. Thus, we first prove the statement for dimension 2. Suppose  $G(q)$  is a complete graph with at least 4 nodes. Let  $H(q)$  be the subgraph of  $G(q)$  induced by the set of vertices  $U_4 = \{1, 2, 3, 4\}$  (i.e. the set of all edges of  $G(q)$  with both of their endpoints in  $U_4$ ). Then,  $H(q) = K_4$  is a complete graph and all nodes of  $H(q)$  are neighbors of each other. This means that nodes 1, 2, 3 together make an equilateral triangle  $\mathcal{T} = \triangle 123$  with a side length of  $d$ . There are two scenarios: i) node  $j = 4$  is inside triangle  $\mathcal{T}$  including its boundaries and ii) node  $j = 4$  is outside  $\mathcal{T}$ . In case i), the distance between  $j$  and at least one of the three other vertices is less than  $d$ . According to the second scenario,  $j$  is outside of  $\mathcal{T}$  and within a distance  $d$  from all vertices of  $\mathcal{T}$ . In particular,  $j$  is within a distance  $d$  from nodes 1, 2. Let  $h = \sqrt{3}d/2$  be the height of an equilateral triangle with a side length of  $d$ . Then, the distance between  $j$  and node 3 is either zero, or  $2h > d$ . In either case,  $j$  cannot be equally distanced from all vertices of  $\mathcal{T}$ . Thus,  $H(q)$  cannot be a complete graph. This is a contradiction and the result follows for dimension 2.

The proof for dimension  $m = 3$  is very similar and the complete proof is omitted. Here is a sketch of the proof:  $G(q)$  is a net with at least  $n = 5$  nodes and  $H(q)$  is a subgraph of  $G(q)$  induced by the set of nodes  $U_5 = \{1, 2, 3, 4, 5\}$ . Moreover, the triangle  $\mathcal{T}$  is replaced by an equilateral tetrahedron (or pyramid)  $\mathcal{P}$  on four vertices 1, 2, 3, 4. Node  $j = 5$  is either inside  $\mathcal{P}$ , or outside  $\mathcal{P}$ . In both cases,  $j$  cannot be within a distance  $d$  from all the vertices of the pyramid  $\mathcal{P}$ . For an equilateral tetrahedron, the height satisfies  $h = \sqrt{6}d/3$  and  $2h > d$ .  $\square$

## B Nonlinear Navigational Feedback: Coupled Normal Forms

Let  $f_\gamma(q, p, q_r, p_r)$  be a *nonlinear navigational feedback* and define  $h_0(q, p, q_r, p_r) = \text{Ave}(f_\gamma(q, p, q_r, p_r))$  and  $g_0(q, p, q_r, p_r) = f_\gamma(q, p, q_r, p_r) - \mathbf{1}_n \otimes h_0(q, p, q_r, p_r)$ . Notice that  $\text{Ave}(g_0(q, p, q_r, p_r)) = 0$ , for all  $q, p, q_r, p_r$ . Then, we have

$$f_\gamma(q, p, q_r, p_r) = g_0(q, p, q_r, p_r) + \mathbf{1}_n \otimes h_0(q, p, q_r, p_r). \quad (80)$$

Define the maps  $g_1, h_1$  as

$$\begin{aligned} g_1(x, v, q_c, p_c, q_r, p_r) &:= g_0(x + \mathbf{1}_n \otimes q_c, v + \mathbf{1}_n \otimes p_c, q_r, p_r) \\ h_1(x, v, q_c, p_c, q_r, p_r) &:= h_0(x + \mathbf{1}_n \otimes q_c, v + \mathbf{1}_n \otimes p_c, q_r, p_r) \end{aligned} \quad (81)$$

Then, the normal forms of structural and translational dynamics are no longer decoupled and can be expressed as

$$\begin{cases} \dot{x} &= v \\ \dot{v} &= -\nabla V(x) - \hat{L}(x)v + g_1(x, v, q_c, p_c, q_r, p_r) \end{cases} \quad (82)$$

and

$$\begin{cases} \dot{q}_c &= p_c \\ \dot{p}_c &= h_1(x, v, q_c, p_c, q_r, p_r) \end{cases} \quad (83)$$

This demonstrates that the decomposition of structural and translational dynamics is solely due to linearity of the navigational feedback. Clearly, the stability analysis of the coupled normal forms of structural and translational dynamics is more challenging than the decoupled version of the normal forms. An example of a nonlinear navigational controller is the following uniformly bounded feedback

$$f_i^\gamma(q_i, p_i, q_r, p_r) = -\text{sat}(c_1(q_i - q_r) + c_2(p_i - p_r)), \quad c_1, c_2 > 0. \quad (84)$$

To implement this feedback, each agent needs to know its own state and the state of its  $\gamma$ -agent, thus the feedback in (84) is a *distributed navigational feedback*. The map  $\text{sat}(\cdot)$  is an  $m$ -dimensional saturation function given by

$$\text{sat}(z) = \begin{cases} z, & \|z\| \leq 1, \\ z/\|z\|, & \|z\| > 1. \end{cases} \quad (85)$$

Even for the case of  $n = 2$  agents,  $h_0(q, p, q_r, p_r) = \text{Ave}(f_\gamma(q, p, q_r, p_r))$  depends on both  $(x, v)$  and  $(q_c, p_c)$ . Notice that for any choice of  $c_1, c_2 > 0$ , there exists a local neighborhood  $\mathcal{N}_{lin}$  of  $(\mathbf{1}_n \otimes q_r, \mathbf{1}_n \otimes p_r)$  in  $\mathbb{R}^{mn} \times \mathbb{R}^{mn}$  such that

$$\|c_1(q_i - q_r) + c_2(p_i - p_r)\| \leq 1, \quad \forall i, \forall (q, p) \in \mathcal{N}_{lin}$$

and  $f_\gamma(q, p, q_r, p_r)$  acts as a linear controller. Thus, as long as the center of group is near  $(q_r, p_r)$ , the structural dynamics and navigational dynamics are decoupled with the controller in (84).

## References

- [1] I. Akyildiz, W. Su, Y. Sankarasubramniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, pages 102–114, August 2002.
- [2] B. Bollobás. *Modern Graph Theory*, volume 184 of *Graduate Texts in Mathematics*. Springer-Verlag, 1998.
- [3] D. E. Chang and J. Marsden. Gyroscopic Forces and Collision Avoidance. *the Conference in Honor of A. J. Krener's 60th Birthday*, 2002.
- [4] D. E. Chang, S. Shadden, J. Marsden, and R. Olfati-Saber. Collision Avoidance for Multiple Agent Systems. *Proc. of the IEEE Conf. on Decision and Control*, Dec. 2003.
- [5] J. Cortes and F. Bullo. Coordination and geometric optimization via distributed dynamical systems. *The SIAM Journal on Control and Optimization (to appear)*, May 2003.
- [6] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *The IEEE Trans. on Robotics and Automation*, 20(2):243–255, April 2004.
- [7] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, 2000.
- [8] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: scalable coordination in sensor networks. *Proc. of Mobile Computing and Networking*, pages 263–270, 1999.
- [9] A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *The IEEE Transactions on Automatic Control (to appear)*, 2004.
- [10] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305, 1973.
- [11] V. Gazi and K. M. Passino. Stability analysis of swarms. *IEEE Trans. on Automatic Control*, 48(4):692–697, April 2003.
- [12] C. Godsil and G. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. Springer, 2001.
- [13] W. R. Heinzelman, J. Kulik, and H. Balakrishnan. Adaptive protocols for information dissemination in wireless sensor networks. *Proc. of Mobile Computing and Networking*, pages 174–185, 1999.
- [14] D. Helbing, I. Farkas, and T. Vicsek. Simulating dynamical features of escape panic. *Nature*, 407:487–490, 2000.
- [15] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1987.
- [16] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile agents using nearest neighbor rules. *The IEEE Trans. on Automatic Control*, 48(6):988–1001, June 2003.
- [17] E. W. Justh and P. S. Krishnaprasad. A simple control law for UAV formation flying. Technical Report 2002–38, Institute for Systems Research, University of Maryland, College Park, MD, 2002.
- [18] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *Int. Journal of Robotics Research*, 5(1):90–98, 1986.
- [19] E. Klavins. Communication Complexity of Multi-Robot Systems. *The 5th International Workshop on the Algorithmic Foundations of Robotics*, Dec. 2002.
- [20] J.-C. Latombe. *Robot motion planning*. Kluwer Academic Publishers, 1991.
- [21] N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials, and coordinated control of groups. *Proc. of the 40th IEEE Conference on Decision and Control*, pages 2968–2973, 2001.
- [22] Y. Liu, K. M. Passino, and M. M. Polycarpou. Stability analysis of  $M$ -dimensional asynchronous swarms with a fixed communication topology. *IEEE Trans. On Automatic Control*, 48(1):76–95, January 2003.
- [23] M. Mesbahi. State-dependent graphs. *Proc. of the 42nd IEEE Conf. on Decision and Control*, 3:3058–3063, Dec. 2003.
- [24] M. Mesbahi. On state-dependent dynamic graphs and their controllability properties. *Submitted (preprint)*, 2004.
- [25] A. Mogilner and L. Edelstein-Keshet. Spatio-temporal order in populations of self-aligning objects: formation of oriented patches. *Physica D*, 89:346–367, 1996.

- [26] A. Mogilner and L. Edelstein-Keshet. A non-local model for a swarm. *J. Math. Biology*, 38:534–570, 1999.
- [27] L. Moreau. Leaderless coordination via bidirectional and unidirectional time-dependent communication. *Proc. of the 42nd IEEE Conference on Decision and Control*, 2004.
- [28] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. *IEEE Trans. on Automatic Control (to appear)*, August 2004.
- [29] A. Okubo. Dynamical aspects of animal grouping: swarms, schools, flocks. and herds. *Adv. Biophysics*, 22:1–94, 1986.
- [30] R. Olfati-Saber. A unified analytical look at Renoldys flocking rules. Technical Report 2003–014, California Institute of Technology, Control and Dynamical Systems, Pasadena, California, September 2003, <http://resolver.caltech.edu/CaltechCDSTR:2003.014>.
- [31] R. Olfati-Saber. Flocking with Obstacle Avoidance. Technical Report 2003–006, California Institute of Technology, Control and Dynamical Systems, Pasadena, California, February 2003, <http://resolver.caltech.edu/CaltechCDSTR:2003.006>.
- [32] R. Olfati-Saber and R. M. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. *The 15th IFAC World Congress*, June 2002.
- [33] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *The IEEE Transactions on Automatic Control (to appear)*, Sep. 2004.
- [34] J. K. Parrish, S. V. Viscido, and D. Grunbaum. Self-organized fish schools: an examination of emergent properties. *Biol. Bull.*, 202:296–305, 2002.
- [35] B. L. Partridge. Rigid definitions of schooling behavior are inadequate. *Animal Behaviour*, 30:298–299, February 1982.
- [36] B. L. Partridge. The structure and function of fish schools. *Scientific American*, 246(6):114–123, 1982.
- [37] B. L. Partridge. The chorus-line hypothesis of maneuver in avian flocks. *Nature*, 309:344–345, 1984.
- [38] T. J. Pitcher, B. L. Partridge, and C. S. Wardle. A blind fish can school. *Scientific American*, 194(4268):963–965, 1976.
- [39] W. Ren and R. W. Beard. Consensus of Information Under Dynamically Changing Interaction Topologies. *Proc. of the American Control Conference*, 2004.
- [40] C. W. Reynolds. Flocks, herds, and schools: a distributed behavioral model. *Computer Graphics (ACM SIGGRAPH '87 Conference Proceedings)*, 21(4):25–34, July 1987.
- [41] C. W. Reynolds. Steering behaviors for autonomous characters. In *Proc. of Game Developers Conference*, pages 763–782. Miller Freeman Game Group, San Francisco, CA, 1999.
- [42] C. W. Reynolds. Interaction with a group of autonomous characters. In *Proc. of Game Developers Conference*, pages 449–460. CMP Game Media Group, San Francisco, CA, 2000.
- [43] G. Ribichini and E. Frazzoli. Efficient coordination of multiple-aircraft systems. *The 42nd IEEE Conf. on Decision and Control*, 1:1035–1040, Dec. 2003.
- [44] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Trans. on Robotics and Automation*, pages 501–518, 1992.
- [45] R. O. Saber and R. M. Murray. Consensus protocols for networks of dynamic agents. *Proc. of the American Control Conference*, 2:951–956, June 2003.
- [46] R. O. Saber and R. M. Murray. Flocking with Obstacle Avoidance: Cooperation with Limited Communication in Mobile Networks. *Proc. of the 42nd IEEE Conference on Decision and Control*, 2:2022–2028, Dec. 2003.
- [47] E. Shaw. Fish in schools. *Natural History*, 84(8):40–45, 1975.
- [48] N. Shimoyama, K. Sugawara, T. Mizuguchi, Y. Hayakawa, and M. Sano. Collective motion in a system of motile elements. *Physical Review Letters*, 76(20):3870–3873, 1996.
- [49] S. H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, 2001.
- [50] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. Flocking in fixed and switching networks. *Submitted to Automatica*, July 2003.
- [51] J. Toner and Y. Tu. Flocks, herds, and schools: A quantitative theory of flocking. *Physical Review E*, 58(4):4828–4858, October 1998.
- [52] T. Vicsek. A question of scale. *Nature*, 411:421–421, May 2001.
- [53] T. Vicsek, A. Cziróók, E. Ben-Jacob, and O. Cohen, I. Shochet. Novel type of phase transition in a system of self-derived particles. *Physical Review Letters*, 75(6):1226–1229, August, 1995.